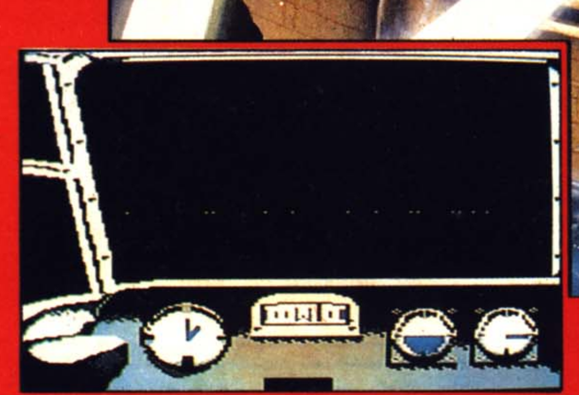

INPUT
Publicación práctica para usuarios de MSX
Revista mensual 1986
Año 1 - Número 8 Precio 375 Ptas.
MSX
Los secretos del chip de video
Calendario/agenda de trabajo
Todo sobre los números romanos
Desenreda tus cadenas

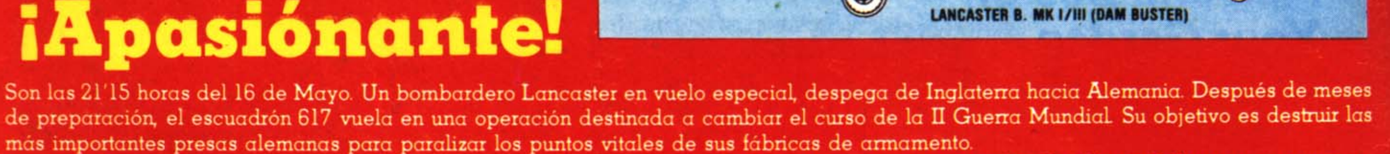
¡ JUEGA EL JUEGO DEL QUE TODOS HABLAN

Distribuido en España por ERBE Software





¡Apasiónante!



Son las 21'15 horas del 16 de Mayo. Un bombardero Lancaster en vuelo especial, despega de Inglaterra hacia Alemania. Después de meses de preparación, el escuadrón 617 vuela en una operación destinada a cambiar el curso de la II Guerra Mundial. Su objetivo es destruir las más importantes presas alemanas para paralizar los puntos vitales de sus fábricas de armamento.

Este detallado y auténtico simulador te permite ocupar los puestos de: **Piloto, Ingeniero de vuelo, Artillero delantero y trasero, Bombardero** y **Navegante.** Volarás a través del Canal de la Mancha y Europa intentando evitar a los temibles ME-110 alemanes, zeppelines, focos antiaéreos y todos los demás peligros a los que se enfrentó el comando Inglés.

AÑO 1 NUMERO 8



DIRECTOR:
Alejandro Diges
DIRECTOR TECNICO:
Roberto Menéndez
COORDINADOR EDITORIAL:
Francisco de Molina
DISEÑO GRAFICO:
Tomás López
COLABORADORES:
Antonio Taratiel, Luis R. Palencia, Francisco Tórtola, J. A. Feberero, Esther de la Cal, Ernesto del Valle, Equipo Molisoft, Javier Portillo.

INPUT MSX es una publicación de PLANETA-DE AGOSTINI, S. A.

GERENTE DIVISION DE REVISTAS:
Angel Sabat

PUBLICIDAD: José Real-Grupo Jota
Madrid: c/ Gral. Varela, 35, 3.º-11
Teléf. 270 47 02/03
Barcelona: Avda. de Sarriá, 11-13, 1.º
Teléf. 250 23 99

FOTOMECANICA: Ochoa, S. A.

COMPOSICION: EFCA, S. A.

IMPRESION: Sirven Grafic
C/ Gran Vía, 754-756. 08013 Barcelona
Depósito legal: B. 38.113-1986

SUSCRIPCIONES: EDISA,
López de Hoyos, 141. 28002 Madrid
Teléf. (91) 415 97 12

REDACCION:
Paseo de la Castellana, 93, 14.ª
28046 Madrid. Teléf. 456 54 13

DISTRIBUIDORA
R.B.A. PROMOTORA DE EDICIONES, S. A.
Travesera de Gracia, 56. Edificio Odiseus.
08006 Barcelona.

El precio será el mismo para Canarias que para la Península y en él irá incluida la sobretasa aérea.

INPUT MSX es una publicación controlada por



INPUT MSX

SUMARIO

EDITORIAL — 4

BUZON — 6

APLICACIONES
TIENES UNA CITA — 10

PARTICIPA
PARMSX — 16

PROGRAMACION
NUMEROS ROMANOS — 28
DESENREDA TUS CADENAS — 46

EN TORNO AL SISTEMA
LA MEMORIA DE VIDEO DE MSX, EL VDP — 40

INTELIGENCIA ARTIFICIAL
UN PROGRAMA QUE JUEGA A LAS 4 EN RAYA (II) — 52

REVISTA DE SOFTWARE — 58

EL ZOCO — 64

LIBROS — 66

PROGRAMACION DE JUEGOS (COLECCIONABLE) — 31
COMPLETA TU AVENTURA
AVENTURAS. EL SIGUIENTE PASO

LA TRANSICION QUE NOS VIENE

Llegan las entrañables fechas de la Navidad, que para el sector informático representan una reactivación de las ventas, sea de *hardware* o, con mayor intensidad, del *software* de divertimento. Nos encontramos en momentos de transición del mercado. El 87 será el momento de expansión del segmento **PC** y compatribles. La principal razón obedece a la brusca bajada de precios y la incorporación de nuevos fabricantes de compatibles. Tal es el caso de **Amstrad**, que con su esperado **PC 1512** logrará disparar la demanda por parte de los usuarios domésticos. Baja el listón para acceder a sistemas que hasta hace pocos meses estaban considerados como coto de los profesionales y empresas.

En el otro extremo se sitúa el ordenador netamente destinado a juegos. Los nuevos precios mas bajos, como en el caso del propio **Spectrum +2** y los **MSX**, conseguirán que esta clase de sistemas sean ámpliamente adoptados en los hogares como un artículo de consumo mas.

Grandes esperanzas se han puesto en el prometedor 1987.

Por nuestro lado os deseamos **Felíz Navidad** a todos los que compartís estas páginas mes a mes con nosotros.

LOS MEJORES DE INPUT

Hemos pensado que es interesante disponer de un ***ranking*** que ponga en claro, mes a mes, cuáles son los programas preferidos de nuestros lectores. Para ello, es obligado preguntaros directamente y tener así el mejor termómetro para conocer vuestras preferencias. Podéis votar por cualquier programa aunque no haya sido comentado todavía en **INPUT**.

El resultado de las votaciones será publicado en cada número de **INPUT**.

Entre los votantes sortearemos 10 cintas de los títulos que pidáis en vuestros cupones.

Nota: No es preciso que cortéis la revista, una copia hecha a máquina o una simple fotocopia sirven.

Enviad vuestros votos a: **LOS MEJORES DE INPUT** P.° de la Castellana, 93. Planta 14. 28046 Madrid

ELIGE TUS PROGRAMAS

Primer título elegido ____________________

Segundo título elegido ____________________

Tercer título elegido ____________________

Programa que te gustaría conseguir ____________________

Qué ordenador tienes ____________________

Nombre ____________________

1.er Apellido ____________________

2.° Apellido ____________________

Fecha de nacimiento ____________________

Teléfono ____________________

Dirección ____________________

Localidad ____________________

Provincia ____________________

INPUT MSX N.° 8


KONAMI TE PRESENTA
TRES AVENTURAS DE PELICULA
SERMA
¿HAS VISTO LAS NOVEDADES DE KONAMI?
NEMESIS, TU MEJOR JUEGO DEL ESPACIO EN MSX.
SE TU EL PROTAGONISTA DE LA GUERRA DE LAS GALAXIAS.
Y HABLANDO DE PELICULAS, ¿TE ACUERDAS DE
LOS GOONIES?, AHORA TE TOCA A TI RESCATAR A TUS
COMPAÑEROS ENCERRADOS EN LAS CUEVAS, MIENTRAS
TE DESHACES DE SUS PELIGROSOS ENEMIGOS.
Y POR SI TE GUSTAN LAS AVENTURAS DE TODO TIPO:
NIGHTMARE, PARA QUE BUSQUES A TU PRINCESA
POR LAS MIL Y UNA PANTALLAS.
NEMESIS
Knightmare
魔城伝説
©Konami 1986
The GOONies
グーニーズ
©Konami 1986
TM
P.V.P : 5.200 pts.
RECORTA Y ENVIA ESTE CUPON A: SERMA. C/. CARDENAL BELLUGA, 21. 28028 MADRID. TELFS. 256 21 01./02.
TITULO:
NOMBRE Y APELLIDOS:
DIRECCION:
CODIGO POSTAL:
POBLACION:
PROVINCIA:
FORMA DE PAGO:
ENVIO TALON BANCARIO □ - CONTRA REEMBOLSO □

EL BUZON DE INPUT



Estoy escribiendo un programa de dibujo y al acabarlo me faltan las instrucciones necesarias para grabar una pantalla.

Es decir, grabar un dibujo hecho con este programa de tal forma que no se grabe el programa, con el fin de aprovechar la pantalla gráfica en algún otro tipo de listados, de juegos por ejemplo. Estoy haciendo un programa de diseño gráfico.

Santiago Vilanova Piñeiro
Lugo

Lo más probable es que estés trabajando en SCREEN 2, en la pantalla de alta resolución. En este caso, para almacenar la pantalla gráfica en cassette *o* diskette, *tienes que guardar los valores de los bytes de las tablas 10, 11 y 12 (tablas de la VRAM). Para ello puedes diseñarte una rutina en BASIC o en lenguaje máquina (según la velocidad que quieras conseguir) que transfiera de la VRAM al* cassette, *byte a byte las tablas completas.*

Otra posibilidad que tienes, es la de transferir primero las tablas a la memoria RAM y utilizar a continuación el comando BSAVE. Este procedimiento es el que comentábamos en el número 4 de ***INPUT***, *al hablar de SCREEN 2. La rutina que ofrecíamos es la siguiente:*

```
10 CLEAR 200,48899!
20 FOR I=1 TO 74
30 READ A$
40 POKE 62000!+I,
   VAL("&H"+A$)
50 NEXT I
60 DEF USR0=62001!
70 DEF USR1=62038!
10000 DATA 21,00,00,01,00,18,
      11,04,BF,CD,59,00,21,00
      ,20,01,00,18,11,0A,D7,C
      D,59,00,21,00,18,01,00,
      03,11,10,EF,CD,59,00,C9

10010 DATA 21,04,BF,01,00,18,
      11,00,00,CD,5C,00,21,0A
      ,D7,01,00,18,11,00,20,C
      D,5C,00,21,10,EF,01,00,
      03,11,00,18,CD,5C,00,C9
```

Puedes añadirla al principio de tu programa, cambiando los números de línea si es necesario. Después tendrás que introducir un par de líneas adicionales que son:

```
BSAVE "CAS:Nombre"
,48900,61968
```

para almacenar la pantalla gráfica

```
BLOAD "CAS:Nombre"
```

para cargar la pantalla gráfica.

La rutina, en código máquina, es muy rápida. Transfiere la pantalla gráfica en sólo un quinto de segundo.



¡Hola! Soy suscriptor de su revista y usuario de un SVI 738. **Cuando intento salvar en** ***diskette*** **los programas que dáis en la revista y que incluyen rutinas código máquina, ocurre que me quedo colgado casi todas las veces. Una sola vez he conseguido grabar una rutina con BSAVE y direcciones de inicio y final, pero a continuación no puedo volver a cargarla en memoria y hacer un listado. Acudo al** MSX-DOS **y puedo listarlo con TYPE, pero aparecen las conocidas figurillas chinas en casi todo el programa y sólo algunas líneas en castellano-BASIC.**

¿Podríais decirme qué es lo que ocurre? ¿Cómo podré conservar esas rutinas y utilizarlas?

Miguel Regaira Martínez
Navarra

Cuando incluimos rutinas en código máquina en alguno de los listados BASIC, lo hacemos a través de las sentencias DATA del propio listado. Es decir, las rutinas código máquina van incluidas en el listado BASIC, y para hacerlas funcionar no tienes más que copiar este listado y seguir las instrucciones que te damos en el artículo. Otra cosa muy distinta son los listados en lenguaje ensamblador de las rutinas código máquina. Estos listados, incluidos para aquellos interesados en la rutina original en lenguaje ensamblador, sólo pueden ser tecleados y salvados en diskette *o* cassette *a través de algunos de los programas ensambladores del mercado.*

Probablemente habrás intentado introducir estos listados desde el BASIC y de ahí vienen todos tus problemas. Lo que tienes que hacer es conseguir un buen ensamblador si tienes intención de trabajar directamente con rutinas código máquina.

Otra posibilidad que te queda abierta para salvar las rutinas ensamblador con BSAVE es la de utilizar los programas BASIC para cargarlas en memoria. Una vez cargadas y conociendo las direcciones inicial y final puedes utilizar BSAVE sin ningún problema.

Lo que nos comentas de TYPE, desde el ***MSX-DOS***, *es totalmente normal teniendo en cuenta que intentas listar por pantalla un listado BASIC* tokenizado. *Para poder listar un programa BASIC con este comando, tienes que haberlo salvado previamente como fichero ASCII, empleando la sintaxis: SAVE «Nombre»,A.*



EL BUZON DE INPUT

Hola amigos, desde hace bastante tiempo estoy interesado con el temporizador incorporado en mi Sony MSX, **hacer un programa con el cual visualizar en la pantalla un reloj en el que aparezcan las horas, los minutos y los segundos. Hasta ahora no lo he conseguido y por eso os escribo, para saber si esto es posible. Si lo es, me gustaría que publicarais dicho programa.**

Otra duda que tengo es la forma de escribir con dos o más colores a la vez.

Victor José Ganan Balza
Tenerife

*Suponemos que tienes un **MSX** de la primera generación y que el temporizador al que te refieres es el que puede leerse a través de la variable TIME. Para realizar el programa que nos comentas, que por supuesto se puede realizar, tienes que tener en cuenta que esta variable se incrementa en uno cada cincuentavo de segundo (es decir se actualiza 50 veces cada segundo). Puedes modificar su valor en cualquier momento sin más que escribir TIME=*nuevo valor. Por último tienes que tener en cuenta que alcanza un valor máximo de 65535 después del cual vuelve a cero y comienza de nuevo la cuenta.

Si quieres trabajar en código máquina, te diremos que las direcciones de memoria que corresponden al reloj son la FC9EH y la FC9FH.

La rutina que sigue te presenta en pantalla el valor de esta variable en segundos y minutos. Los segundos se obtienen dividiendo el valor de TIME entre 50 y los minutos dividiendo este valor entre 60. Para pasar a horas, tendrás que actualizar una variable cada vez que se cumplan 60 minutos. Esperamos que con estas ideas y la rutina que te ofrecemos, seas tu mismo capaz de desarrollar el programa que necesitas.

```
10 CLS:TIME=0
20 T=TIME
30 SE=(T/50)MOD60
40 MI=INT(T/3000)
50 LOCATE0,0
60 PRINT USING"##";MI;
   :PRINT":"
70 LOCATE4,0
80 PRINT USING"##";SE
90 GOTO 20
```

En cuanto al escribir con dos o más colores a la vez, vas a tener que trabajar en un modo de pantalla que no sea SCREEN 0. Por ejemplo en SCREEN 2 o en SCREEN 1. Consulta algún buen manual del BASIC ***MSX***.



Me gustaría saber si se pueden enviar programas en código máquina presentando solamente las líneas DATA del cargador BASIC, o si hace falta un listado ensamblador.

Alberto Alvarez Besada
Pontevedra

A ser posible conviene que envíes las dos cosas. El cargador BASIC, con los valores de la rutina en sentencias DATA, es para que aquellos que no dispongan de un programa ensamblador puedan disfrutar de la misma. Por otro lado, el listado ensamblador será una gran ayuda para todos los que quieran analizar la rutina y entender su funcionamiento.



Quisiera saber si se ha publicado algún libro con la ROM de los ordenadores de la norma MSX **desensamblada, pues se que existe para otros ordenadores. Os agradezco por adelantado la respuesta.**

Juan Manuel Marsa Fauste
Barcelona

*Según nuestras noticias, no se ha publicado en nuestro país ningún libro de las características que nos comentas. Sabemos que se puede encontrar alguna copia del libro del «Estándar **MSX**» de **Microsoft**, en el que se dan algunas pistas sobre las rutinas de ROM. Este libro está en inglés. Asimismo, y publicado en Inglaterra por **Avalon Software**, hay un libro titulado «**MSX**red book» (el libro rojo de **MSX**) en el que se incluye una información muy detallada de las distintas rutinas, variables, etc. de la ROM. Desgraciadamente no parece que nadie se haya decidido a importar o a publicar la traducción de este interesante libro.*



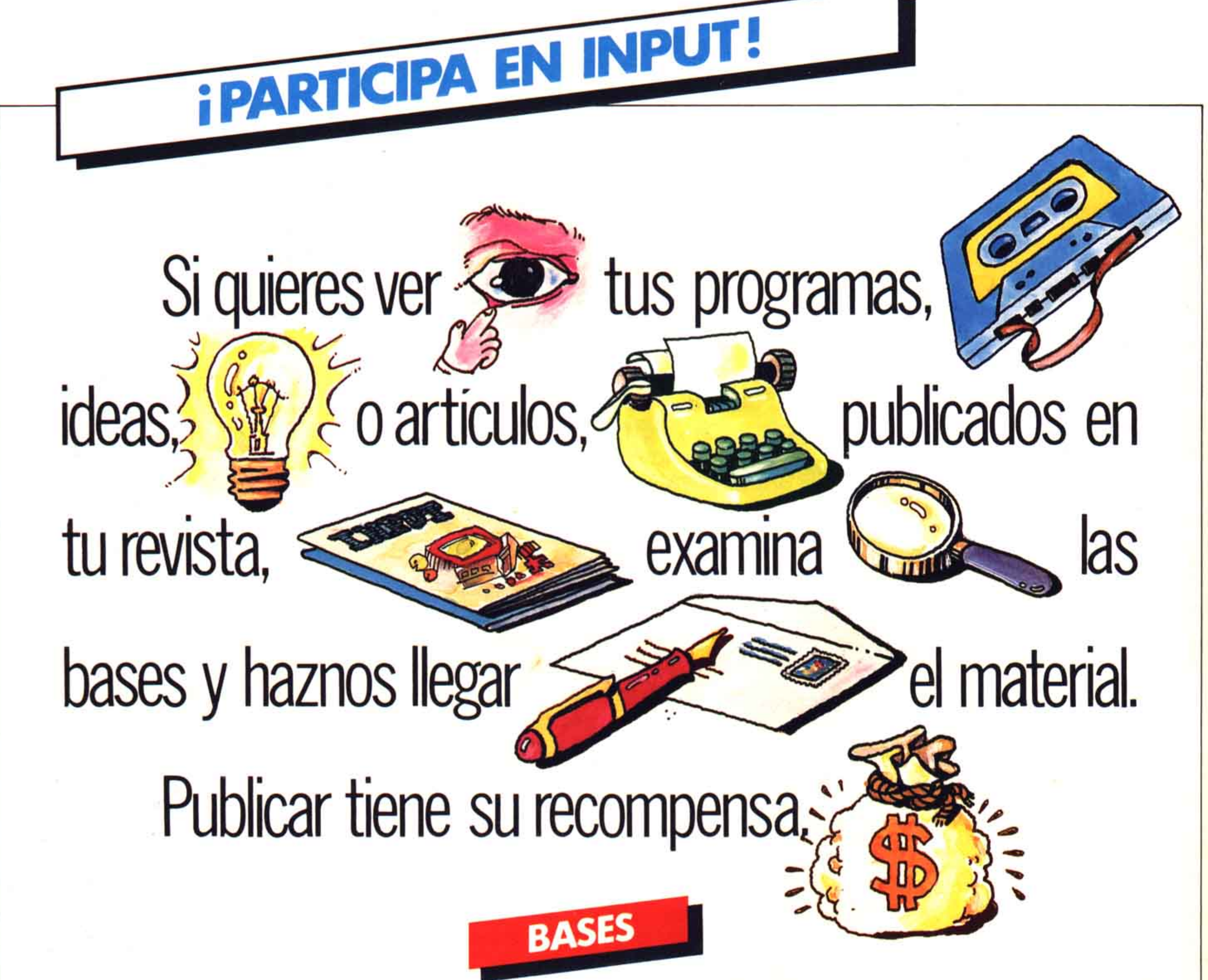
Hola. Me gustaría saber si los fabricantes han lanzado al mercado alguna versión del lenguaje COBOL para MSX. **Me interesaría cualquier información.**

Violeta Lopen Sanz
Madrid

*La empresa **Spectravideo España** comercializa una versión de lenguaje COBOL para **MSX**. Se trata de un compilador de este lenguaje de la firma estadounidense **Ellis Computing** y lleva el nombre de **Nevada** COBOL. Es una versión de excelente calidad y corre bajo el sistema operativo **MSX-DOS** (es decir necesita del empleo de al menos una unidad de* diskettes*). Te recomendamos que entres en contacto con **Spectravideo**. Ellos te aclararán cualquier duda sobre su producto.*



¡PARTICIPA EN INPUT!



BASES

PROGRAMAS: Una vez desarrollado tu programa, que debe ser original y no haber sido enviado a ninguna otra publicación, puedes enviárnoslo aquí grabado en cassette, diskette o microdrive. Es preferible que vaya acompañado por un listado de impresora, pero no es imprescindible.

El programa habrá de venir acompañado por un texto que aclare cuál es su objetivo, el modo de funcionamiento y una explicación del cometido que cumplen las distintas rutinas que lo componen. El texto se presentará en papel de tamaño folio y mecanografiado a dos espacios. No importa que la redacción no sea muy clara y cuidada; nuestro equipo de expertos se encargará de proporcionarle la forma más atractiva posible.

ARTICULOS E IDEAS: Se aplica lo anteriormente dicho para los textos que acompañan a los programas; es decir, conviene detallar al máximo lo que desees que aparezca publicado en la revista, de la manera que te gustaría que otra persona hubiera explicado eso mismo.

UN JURADO propio decidirá en cada momento qué colaboraciones reúnen los requisitos adecuados para su publicación, y evaluará la cuantía del premio en metálico al que se hagan acreedoras.

No olvidéis indicar claramente para qué ordenador está preparado el material, así como vuestro nombre y dirección y, cuando sea posible, un teléfono de contacto. Entre todos los trabajos recibidos durante cada mes SORTEAREMOS:

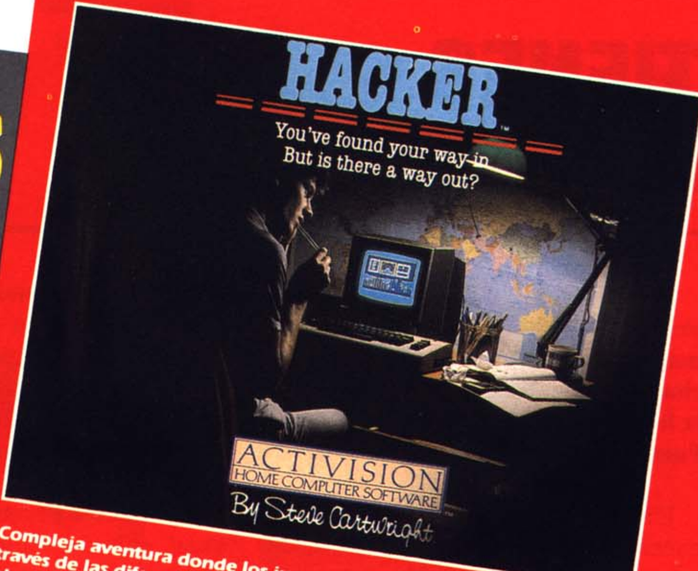
- **Un premio de 50.000 ptas.**
- **Un premio de 25.000 ptas.**
- **Un premio de 10.000 ptas.**
  en material microinformático a elegir por los afortunados.

¡No os desaniméis!, por muy simples o complejas que puedan parecer vuestras ideas, todas serán revisadas con el máximo interés.

INPUT MSX
Paseo de la Castellana, 93. Planta 14
28046 MADRID

NOTA: INPUT no se responsabiliza de la devolución del material que no vaya acompañado por un sobre adecuado con el franqueo correspondiente.

TE PRESENTAMOS LOS IMPRESCINDIBLES PARA TU ORDENADOR



Compleja aventura donde los jugadores deben buscar a través de las diferentes pistas y problemas como resolver el misterio.

CSAM





El más espectacular KARATE hasta ahora nunca visto. Practica las artes marciales en distintos escenarios del Mundo.
"ZAPP" dijo que era el mayor desafío en juegos de este tipo.

C M



El más reciente avance técnico. Vd. puede realizar una jornada completa dentro de una cápsula espacial desde que se levanta de la superficie de la tierra y acude a un encuentro en el espacio, hasta que aterriza nuevamente. Comprueba tu habilidad.

CSM

Disponibles para:

COMMODORE	C
SPECTRUM	S
AMSTRAD	A
MSX	M

EN TIENDAS ESPECIALIZADAS Y GRANDES ALMACENES, O DIRECTAMENTE POR CORREO O TELEFONO A: **PROEIN, S.A.**

Distribuido en Cataluña por: DISCOVERY INFORMATIC C/. Arco Iris, 75 - BARCELONA - Tels. 256 49 08 / 09

Velázquez, 10 - 28001 Madrid - Tels. (91) 276 22 08/09

TIENES UNA CITA

- CALENDARIO ANUAL
- DIA DEL AÑO Y DE LA SEMANA
- SIGNOS DEL ZODIACO
- DIAS ENTRE FECHAS

¿No serás una de esas personas que siempre se olvida del cumpleaños de su madre, o que recuerda que tiene una cita con el dentista dos días después de la fecha en que debiera haber ido? ¿Si? Pues toma nota de este programa.

Una maniobra que todos nosotros hacemos con cierta frecuencia es consultar un calendario. Seguramente tenemos uno pegado en la pared, o sobre la mesa, o en la agenda.

Lo normal es disponer del calendario correspondiente al año en curso, aunque algunas veces, sobre todo en las agendas, suele venir también el año anterior e incluso el siguiente. En algunos calendarios aparece información complementaria sobre el número de orden de cada día dentro del año, santoral, períodos astrales, etc.

El problema aparece cuando necesitamos consultar calendarios antigüos o de años venideros. Lo normal es que no dispongamos de ellos.

Esto tendría solución si pudiéramos imprimir un calendario adaptado a nuestras necesidades específicas en el que figurase, además del calendari con el mes deseado, información complementaria tal como fiestas locales o particulares, fechas en que realizar ciertos trabajos, vencimientos de pagos, etc.

Las consultas a los calendarios se hacen para hallar respuesta a preguntas como las siguientes:

- ¿En qué día de la semana comenzó la Guerra Civil Española (18 de Julio de 1936)?

- ¿Cuántos días hemos estado viviendo en Roma (12/3/1985 - 1/8/1986)?

- ¿Qué fecha corresponde a 1318 días a partir de 4 de Mayo de 1725?

- ¿Bajo qué signo del Zodíaco hemos nacido?

En teoría la respuesta no requiere de conocimientos especiales. Basta disponer de abundantes calendarios, papel, lápiz y una calculadora (en último caso podemos contar con los dedos). En la práctica las cosas quizá se nos compliquen un poco, sobre todo si tenemos que manejar fechas muy distantes.

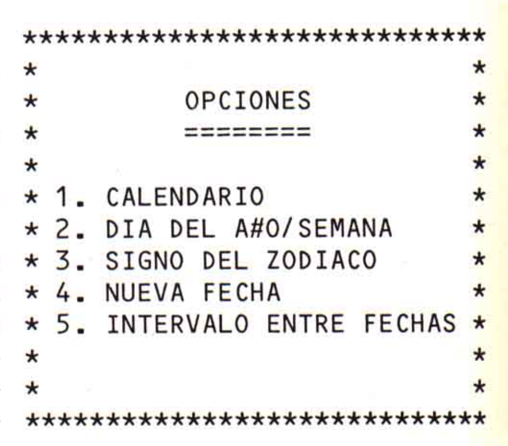
Como en otras muchas facetas de la vida, el ordenador puede prestarnos una excelente ayuda. En este sentido hemos preparado el programa «Manejo del Calendario» cuyo contenido pasamos a comentar. Como anexo figura el correspondiente listado.

Dedicamos también un cierto espacio a aquellos «viciosos de la programación» que desean conocer el por qué de las cosas. No les será difícil modificar el programa y enriquecerlo con temas de su propia cosecha.

MANEJO DEL PROGRAMA

Lo primero que aparece es un menú con cinco opciones:





```
*****************************
*                           *
*         OPCIONES          *
*         ========          *
*                           *
* 1. CALENDARIO             *
* 2. DIA DEL A#O/SEMANA     *
* 3. SIGNO DEL ZODIACO      *
* 4. NUEVA FECHA            *
* 5. INTERVALO ENTRE FECHAS *
*                           *
*                           *
*****************************
```

Opción 1:

El ordenador nos presenta una pantalla base y nos va solicitando sucesivamente los datos del Año y Mes, así como los días que queremos fijar como festivos (hasta cinco fechas por mes) y el texto (hasta 40 caracteres) que deseamos reflejar al pie del calendario. El aspecto de la pantalla será como el siguiente:

```
          CALENDARIO
          ==========

A#O       :1986
MES       :DICIEMBRE
```



```
FIESTAS :8 25
OBSER.  :PLANIFICAR MANTENI-
MIENTOS PRIMER TRIMESTRE 1987

MODIFICAR (S/N)?
```

En este punto podremos rectificar cualquier dato. Al final nos aparecerá:

```
      DICIEMBRE 1986

  D   L   M   X   J   V   S
-----------------------------
:   : 1 : 2 : 3 : 4 : 5 : 6 :
-----------------------------
: 7 : 8 : 9 : 10: 11: 12: 13:
-----------------------------
: 14: 15: 16: 17: 18: 19: 20:
-----------------------------
: 21: 22: 23: 24: 25: 26: 27:
-----------------------------
: 28: 29: 30: 31:   :   :   :
-----------------------------

    FIESTAS : 8  25
    OBSERV. : PREPARA MANTENI
    MIENTO PRIMER TRIMESTRE
    1987

IMPRIMIR (S/N)?
```



Si todo aparece conforme a nuestros deseos podemos proceder a su impresión. La copia de la impresora es un fiel reflejo de lo que aparece en pantalla y por lo tanto debe utilizarse una que siga el estándar **MSX**, ya que la retícula del calendario está realizada con los signos gráficos **MSX** adecuados para que el trazado sea continuo. Si disponemos de otro tipo de impresora deberá modificarse el programa para utilizar otros símbolos (por ejemplo los que aparecen en la presentación anterior).



Opción 2:

Se nos facilita la fecha deseada y el ordenador nos calcula y presenta los resultados de la manera siguiente:

```
FECHA  : 30 / 10 / 1986

MES               :OCTUBRE
DIA DEL A#O       :303
DIA DE LA SEMANA  :JUEVES
SIGNO DEL ZODIACO:ESCORPIO
```

Opción 3:

Mediante esta opción podemos conocer las fechas que cubren el período asignado a cada uno de los símbolos. Una vez tecleado el nombre correspondiente (bastan las tres primeras letras) aparece en pantalla la siguiente información:

```
CAPRICORNIO
===========

Es el signo del zodiaco que
corresponde a los nacidos
entre las fechas:

22 DE DICIEMBRE
20 DE ENERO
```

Opción 4:

Dada una fecha cualquiera el programa nos calcula otra que esté separada de ella una serie de días predefinidos. De la nueva fecha también aparecen los mismos datos de la Opción 2.

Opción 5:

Se trata de la recíproca de la Opción anterior. En este caso el programa nos solicita dos fechas cualesquiera y nos calcula el número de días que las separan. Debido al procedimiento



de cálculo utilizado, el margen de fechas en el que nos podemos mover es: 15 de Octubre de 1582 al 25 de Noviembre del 4046. Entendemos que se trata de un margen suficientemente amplio.

ASPECTOS MAS IMPORTANTES DEL PROGRAMA

Días de la semana:

Como referencia, comenzamos las semanas por Domingo y le identificaremos por día «0». Los nombres de los diversos días los alojamos en la matriz D$() mediante una lectura de los DATAs correspondientes (instrucciones 60,70 y 80).

Meses del año:

El proceso es análogo pero utilizando la matriz M$() (instrucciones 90,100 y 110).

Ambas operaciones se realizan una sola vez, la primera vez que se rueda el programa.

Períodos del Zodíaco:

Para cada período se confeccionan tres tipos de datos que se alojan en los DATAs correspondiente (930-1040). El primer dato es el número formado por el valor del mes multiplicado por 100 más el valor del día, ambos referidos a la fecha baja del período. El segundo dato es análogo pero para la fecha alta. El tercero es el nombre asignado al símbolo.

A la hora de localizar las fechas de un Signo del Zodíaco hemos optado por dar una pasada (340-390) a este grupo de DATAs hasta encontrar el nombre buscado (por comodidad sólo contrastamos las tres primeras letras y el nombre lo debemos fijar con letras mayúsculas). Los DATAs asociados a este nombre nos permitirán reconstruir las fechas. Ej:

Dato 1 = 823, Dato 2 = 923, Dato 3 = «VIRGO»

Dividiendo por 100 y tomando la parte entera conoceremos el número de los meses, y por diferencia el de los días. En este caso 8 y 23, 9 y 23. M$(8) y M$(9) nos darán respectivamente Agosto y Septiembre.

Calendario Perpetuo:

El plato fuerte del programa está indudablemente en la rutina para el cálculo de los datos del «calendario perpetuo» (MANEJO DE FECHAS). No vamos a entrar en los tecnicismos matemáticos en los que está basado el procedimiento de cálculo, simplemente sabemos que partiendo de una fecha determinada (D,M,A) obtenemos el valor del «día juliano» (J) correspondiente, el cual nos sirve como referencia absoluta.

Por diferencia con el día juliano del primer día del año en curso (J0) podemos saber el número de orden de nuestra fecha dentro del año (DA). Evidentemente el procedimiento tiene en cuenta automáticamente los años bisiestos.

Mediante otro sencillo cálculo conoceremos el orden dentro de la semana de la fecha que estamos analizando (DS). El 0 corresponde al domingo y así sucesivamente hasta el 6 para el sábado. Acudiendo a la matriz D$() podremos conocer el nombre asignado a dicho día (D$).

Como final de la rutina, y partiendo de los valores del mes y del día, averiguaremos a que signo del Zodíaco corresponde con ayuda de una sentencia «IF».

Más adelante en el programa existe otra rutina para hacer justo la maniobra opuesta. Partiendo de un valor de día juliano averiguar a que fecha (D, M y A) le corresponde.

Con estas breves pinceladas sobre el programa entendemos les bastará a nuestros lectores para seguir el detalle de cada uno de los pasos.

```
10 'CALENDARIO
20 'INPUT MSX
30 'MOLISOFT OCT.86
40 CLEAR 500:SCREEN 0:WIDTH
   32 :KEY OFF
50 LOCATE 7,2:PRINT"I N P U T
   M S X"
60 LOCATE 7,3:PRINT"=========
   ========"
70 LOCATE 10,5:PRINT"M A N E
   J O ":LOCATE 14,8:PRINT
   "DE": LOCATE 11,11:PRINT
   " FECHAS"
80 LOCATE 8 ,15:PRINT
   "Molisoft Nov.86"
```

```
90 FOR I=1 TO 2000:NEXT I
100 DIM CL$(42):REM DISTRIBUC
    ION DE DIAS EN EL MES
110 DIM F$(5):REM DIAS
    FESTIVOS
120 DIM DS$(6):REM DIAS DE LA
    SEMANA
130 DATA DOMINGO,LUNES,MARTES
    ,MIERCOLES,JUEVES,VIERNES
    ,SABADO
140 FOR L=0 TO 6:READ X$:DS$
    (L)=X$:NEXT L
150 DIM M$(12):REM MESES DEL
    A#O
160 DATA ENERO,FEBRERO,MARZO,
    ABRIL,MAYO,JUNIO,JULIO,AG
    OSTO,SEPTIEMBRE,OCTUBRE,N
    OVIEMBRE,DICIEMBRE
170 FOR L=1 TO 12:READ X$:M$
    (L)=X$:NEXT L
180 GOSUB 1170:REM ORLA MENU
190 LOCATE 9 ,2:PRINT"O P C I
     O N E S"
200 LOCATE 9 ,3:PRINT"=======
    ========"
210  LOCATE 5,6:PRINT "1. CAL
    ENDARIO"
220  LOCATE 5,8:PRINT"2. DIA
    DEL A#O / SEMANA"
230  LOCATE 5,10:PRINT"3. SIG
    NO ZODIACO"
240  LOCATE 5,12:PRINT"4. NUE
    VA FECHA"
250  LOCATE 5,14:PRINT"5. INT
    ERVALO ENTRE FECHAS"
260 P$= INKEY$:IF P$=""
    THEN 260 ELSE P=VAL(P$)
270 IF  P<1 OR P>5 THEN 260
280 ON P GOTO 600,290,380,490
    ,560
290 REM OPCION 2
300 GOSUB 1590:CLS
310 GOSUB 1210:PRINT"FECHA   :
    ";D;"/";M;"/";A:PRINT
320 PRINT "MES..............:
    ";M$:PRINT
330 PRINT "DIA DEL A#O......:
    ";DA:PRINT
340 PRINT "DIA DE LA SEMANA.:
    ";DS$:PRINT
350 PRINT"SIGNO ZODIACO....:
    ";H$
360 GOSUB 1160:PRINT "CUALQUI
    ER TECLA PARA MENU"
370 IF INKEY$="" THEN 370
    ELSE 180
380 REM OPCION 3
390 GOSUB 1160:INPUT "SIGNO D
    EL ZODIACO (letr.may.) ";
    P$
400 RESTORE 1470
410 X=0:FOR L=1 TO 12 :READ X
    1,X2,X$:IF LEFT$(X$,3)=LE
    FT$(P$,3) THEN X=1:L=12
420 NEXT L
430 IF X=0 THEN 180
440 F1=INT(X1/100):F3=INT(X2/
    100)
450 F2=X1-F1*100:
    F4=X2-F3*100
460 CLS:PRINT TAB(3)X$:PRINT
    TAB(3) STRING$(LEN(X$),"=
    "):PRINT:PRINT "   Es el
    Signo del  ZODIACO que":P
    RINT "   corresponde  a
    los  nacidos":PRINT"   en
    tre las fechas :":PRINT:P
    RINT TAB(6)F2;" DE ";M$(F
    1):PRINT:PRINT TAB(6)F4;"
     DE ";M$(F3)
470 GOTO 360
480 REM OPCION 4
490 GOSUB 1590
500 GOSUB 1160:INPUT"( A CUAN
    TOS  DIAS  DE  LA FECHA A
    NTERIOR ";N:IF N<1 THEN 5
    00
510 CLS:PRINT"A";N;" DIA(S) D
    E";D;"/";M;"/";A:PRINT:PR
    INT "TENEMOS  LA  SIGUIEN
    TE ":PRINT
520 GOSUB 1210:GOSUB 1690
530 IF J>1478160! THEN GOSUB
    1850:GOTO 180
540 GOTO 310
550 'INTERVALO ENTRE FECHAS
560 GOSUB 1590:GOSUB 1210:R1=
    D:R2=M:R3=A:R4=J
570 GOSUB 1590:GOSUB 1210:Q1=
    D:Q2=M:Q3=A:Q4=J
580 CLS:PRINT " EL INTERVALO
    ENTRE LAS FECHAS :":PRINT
    :PRINT R1;"/";R2;"/";R3:P
    RINT :PRINT Q1;"/";Q2;"/"
    ;Q3:PRINT:PRINT " ES DE "
    ;Q4-R4;" DIAS"
590 GOTO 360
```

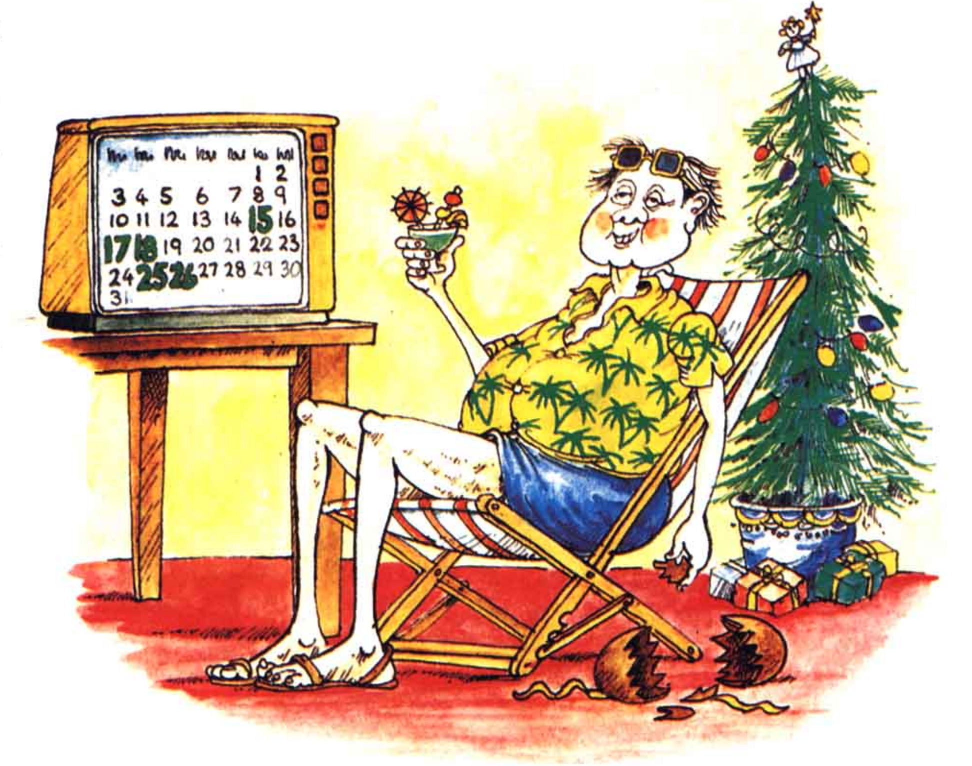


```
600 REM OPCION 1
610 GOSUB 1170
620 LOCATE 11,2:PRINT"CALENDA
    RIO "
630 LOCATE 11,3:PRINT"=======
    === "
640 LOCATE 5,5:PRINT "A#O
     :"
650 LOCATE 5,7:PRINT"MES
    :"
660 LOCATE 5,9 :PRINT"FIESTAS
     :"
670 LOCATE 5,11:PRINT"OBSERV.
     :"
680 FOR I=1 TO 4:F$(I)="":NEX
    T I
690 GOSUB 1160:INPUT"A%0 (158
    2/4046)";A:IF A<1582 OR A
    >4046 THEN 690
700 LOCATE 14,5:PRINT STRING$
    (14," "):LOCATE 14,5:PRIN
    T A
710 GOSUB 1160:INPUT"MES (1-1
    2)";M:IF M<1 OR M>12 THEN
     710
720 LOCATE 14,7:PRINT STRING$
    (14," "):LOCATE 14,7:PRIN
    T M
730 FOR I=1 TO 5
740 GOSUB 1160:D=0:PRINT "(";
    STR$(I);" ) ";:INPUT "DIA
     FESTIVO ";D:IF D=0 THEN
    I=5:GOTO 760
750 IF D<1 OR D>31 THEN 740 E
    LSE F$(I)=STR$(D)
760 LOCATE 14,9:PRINT STRING$
    (14," "):LOCATE 14,9
770 FOR L=1 TO 5:PRINT F$(L);
    :NEXT L
780 NEXT I
790 GOSUB 1160:INPUT"OBSERVAC
    IONES(MAX.40c)";O$:IF LEN
    (O$)>40 THEN 790
800 LOCATE 15,11:PRINT SPACE$
    (15):LOCATE 15,11:IF LEN(
    O$)<=15 THEN PRINT O$ ELS
    E PRINT LEFT$(O$,15)
810 LOCATE 5,13:PRINT SPACE$(
    25):LOCATE 5,13:IF LEN(O$
    )>15 THEN PRINT RIGHT$(O$
    ,LEN(O$)-15)
820 GOSUB 1160:INPUT"MODIFICA
    R S/N";P$:IF P$="S" OR P$
    ="s" THEN 690
830 FOR I=0 TO 42:CL$(I)="":N
    EXT I
840 GOSUB 1760
850 FOR I=DW TO DW+UD-1:CL$(I
    )=STR$ (N+1):N=N+1:NEXT I

860 CLS:L=INT(15-(LEN(M$(M-1)
    )+5)/2):PRINT TAB(L) M$(M
    -1);A:PRINT :PRINT
870 PRINT " D   L   M   X
    J   V   S"
880 PRINT "------------------
    ------------"
890 FOR I=0 TO 35 STEP 7
900 PRINT TAB(0)":"+CL$(I)TAB
    (4)":"+CL$(I+1)TAB(8)":"+
    CL$(I+2)TAB(12)":"+CL$(I+
    3)TAB(16)":"+CL$(I+4)TAB(
    20)":"CL$(I+5)TAB(24)":"+
    CL$(I+6)TAB(28)":"
910 IF CL$(I+7)="" THEN I=35
    :GOTO 930
920 PRINT "------------------
    ------------"
930 NEXT I
940 PRINT"------------------
    ----------":PRINT
950 PRINTTAB(2)"D.Fest. :";
960 FOR I=1 TO 5:PRINT F$(I);
    :NEXT I
970 L=LEN(O$):IF (L>15) THEN
    S$=LEFT$(O$,15):T$=RIGHT$
    (O$,L-15)
980 IF (L<=15) THEN S$=O$:T$=
    ""
990 PRINT:PRINT:PRINTTAB(2)"O
    bserv.:  ";S$:PRINTTAB(2)
    T$
1000 PRINT:INPUT"IMPRIMIR S/N
      ";P$:IF P$<>"S" AND P$<
     >"s" THEN 180
1010 ' IMPRESION CALENDARIO
1020 L=INT(15-(LEN(M$(M-1))+5
     )/2):LPRINT TAB(L) M$(M-
     1);A:LPRINT :LPRINT
1030 LPRINT " D   L   M   X
      J   V   S"
1040 LPRINT "----------------
     --------------"
1050 FOR I=0 TO 35 STEP 7
1060 LPRINT TAB(0)":"+CL$(I)T
     AB(4)":"+CL$(I+1)TAB(8)"
     :"+CL$(I+2)TAB(12)":"+CL
     $(I+3)TAB(16)":"+CL$(I+4
     )TAB(20)":"CL$(I+5)TAB(2
     4)":"+CL$(I+6)TAB(28)":"
1070 IF CL$(I+7)="" THEN I=35
     :GOTO 1090
1080 LPRINT "----------------
     -------------"
1090 NEXT I
1100 LPRINT"-----------------
     ------------":LPRINT
1110 LPRINT TAB(2)
     "D.Fest. :";
1120 FOR I=1 TO 5:LPRINT F$(I
     );:NEXT I
1130 L=LEN(O$):IF (L>15) THEN
     S$=LEFT$(O$,15):T$=RIGHT
     $(O$,L-15)
1140 IF (L<=15) THEN S$=O$:T$
     =""
1150 LPRINT:LPRINT:LPRINT TAB
     (2)"Observ.:   ";S$:LPRIN
     T TAB(2)T$:GOTO 180
1160 LOCATE 0,19:PRINT SPACE$
     (32):LOCATE 0,20:PRINT S
     PACE$(32):LOCATE 0,21:PR
     INT SPACE$(32):LOCATE 0,
     19:RETURN
1170 REM ORLA MENU
1180 CLS:LOCATE 3,0:PRINT STR
     ING$(28,"*")
1190 FOR I=1 TO 15:PRINT TAB(
     3)"*" TAB(30)"*":NEXT I
1200 LOCATE 3,16:PRINT STRING
     $(28,"*"):RETURN
1210 'MANEJO DE FECHAS
1220 A0=A-1:M0=14
1230 A1=(M>2)*(-A)+(M<=2)*(-A
     +1)
1240 M1=(M>2)*(-M-1)+(M<=2)*(
     -M-13)
1250 J=INT(365.25*A1)-INT(A1/
     100)+INT(A1/400)+INT(30.
     6001*M1)+D-478164!
1260 J0=INT(365.25*A0)-INT(A0
     /100)+INT(A0/400)+INT(30
     .6001*M0)+1-478164!
1270 DS=J-7*INT(J/7)
1280 DS$=DS$(DS)
1290 DA=J-J0+1
1300 M$=M$(M)
1310 F=M*100+D
1320 IF F>=321 AND F<=420 THE
     N H$="ARIES":GOTO 1450
1330 IF F>=421 AND F<=521 THE
     N H$="TAURO":GOTO 1450
1340 IF F>=522 AND F<=621 THE
     N H$="GEMINIS":GOTO 1450
1350 IF F>=622 AND F<=722 THE
     N H$="CANCER":GOTO 1450
```

```
1360 IF F>=723 AND F<=822 THE
     N H$="LEO":GOTO 1450
1370 IF F>=823 AND F<=923 THE
     N H$="VIRGO":GOTO 1450
1380 IF F>=824 AND F<=1023 TH
     EN H$="LIBRA":GOTO 1450
1390 IF F>=1024 AND F<=1122 T
     HEN H$="ESCORPIO":GOTO 1
     450
1400 IF F>=1123 AND F<=1221 T
     HEN H$="SAGITARIO":GOTO
     1450
1410 IF F>=1222 AND F<=1231 T
     HEN H$="CAPRICORNIO":
          GOTO 1450
1420 IF F>=101 AND F<=120 THE
     N H$="CAPRICORNIO":GOTO
     1450
1430 IF F>=121 AND F<=220 THE
     N H$="ACUARIO":GOTO 1450

1440 IF F>=221 AND F<=320 THE
     N H$="PISCIS":GOTO 1450
1450 RETURN
1460 ' SIGNO ZODIACO
1470 DATA 321,420,"ARIES"
1480 DATA 421,521,"TAURO"
1490 DATA 522,621,"GEMINIS"
1500 DATA 622,722,"CANCER"
1510 DATA 723,822,"LEO"
1520 DATA 823,923,"VIRGO"
1530 DATA 824,1023,"LIBRA"
1540 DATA 1024 ,1122,
     "ESCORPIO"
1550 DATA 1123,1221,
     "SAGITARIO"
1560 DATA 1222,120,
     "CAPRICORNIO"
1570 DATA 121,220 ,"ACUARIO"
1580 DATA 221,320,"PISCIS"
1590 REM TOMA DE FECHAS
1600 GOSUB 1160:INPUT"(A%O (1
     582-4046)";A:IF A<1582 O
     R A>4046 THEN 1600
1610 GOSUB 1160:INPUT"(MES (1
     -12)";M:IF M<1 OR M>12
          THEN 1610
1620 MS=M:AN=A
1630 GOSUB 1760
1640 GOSUB 1160:INPUT"(DIA";D
     :IF D<1 OR D>31 THEN 164
     0
1650 M=MS:A=AN
1660 IF D>UD THEN  GOSUB 1800
     :GOTO 1590
```




```
1670 F= A*10000+M*100+D:IF F<
     15821015# OR F>40461125#
      THEN GOSUB 1800:GOTO 15
     90
1680 RETURN
1690 J=J+478164!+N
1700 A1#=INT((J-121.5)/365.24
     25#)
1710 M1#=INT((J-INT(365.25*A1
     )+INT(A1/100)-INT(A1/400
     ))/30.6001)
1720 D=J-INT(365.25*A1)+INT(A
     1/100)-INT(A1/400)-INT(3
     0.6001*M1)
1730 M=(M1<14)*(1-M1)+(M1>=14
     )*(13-M1)
1740 A=(M>2)*(-A1)+(M<3)*(-1-
     A1)
1750 RETURN
1760 REM NUMERO MAX DIAS DEL
     MES
1770 D=1:GOSUB 1210:
     Y=J:DW=DS
1780 IF (M<12) THEN M=M+1 ELS
     E M=1:A=A+1
1790 GOSUB 1210:UD=J-Y:N=0:RE
     TURN
1800 REM AVISO DE DATO
     ERRONEO
1810 FOR I=1  TO 10
1820 LOCATE 0,21:PRINT "FECHA
     : ";D;"/";M;"/";A;" ERRO
     NEA"
1830 FOR L=1 TO 200:NEXT L
1840 NEXT I:RETURN
1850 REM AVISO DE DATO
     ERRONEO
1860 CLS:FOR I=1  TO 10
1870 LOCATE 0,21:PRINT "DEMAS
     IADOS DIAS"
1880 FOR L=1 TO 200:NEXT L
1890 NEXT I:RETURN
1900 STOP
9000 SAVE "A:CALEND.BAS"
```

NO OLVIDES EL TELEFONO... ☎

Cuando, por cualquier motivo, nos escribas, no olvides indicar tu número de teléfono. Así nos será más fácil y rápido ponernos en contacto contigo. Gracias.

PARMSX

Empleando algunos conceptos de inteligencia artificial, el programa PARMSX es un conjunto de instrucciones BASIC aparentemente capaces de mantener una conversación. La clave está en un acertado manejo de las instrucciones de tratamiento de cadenas.

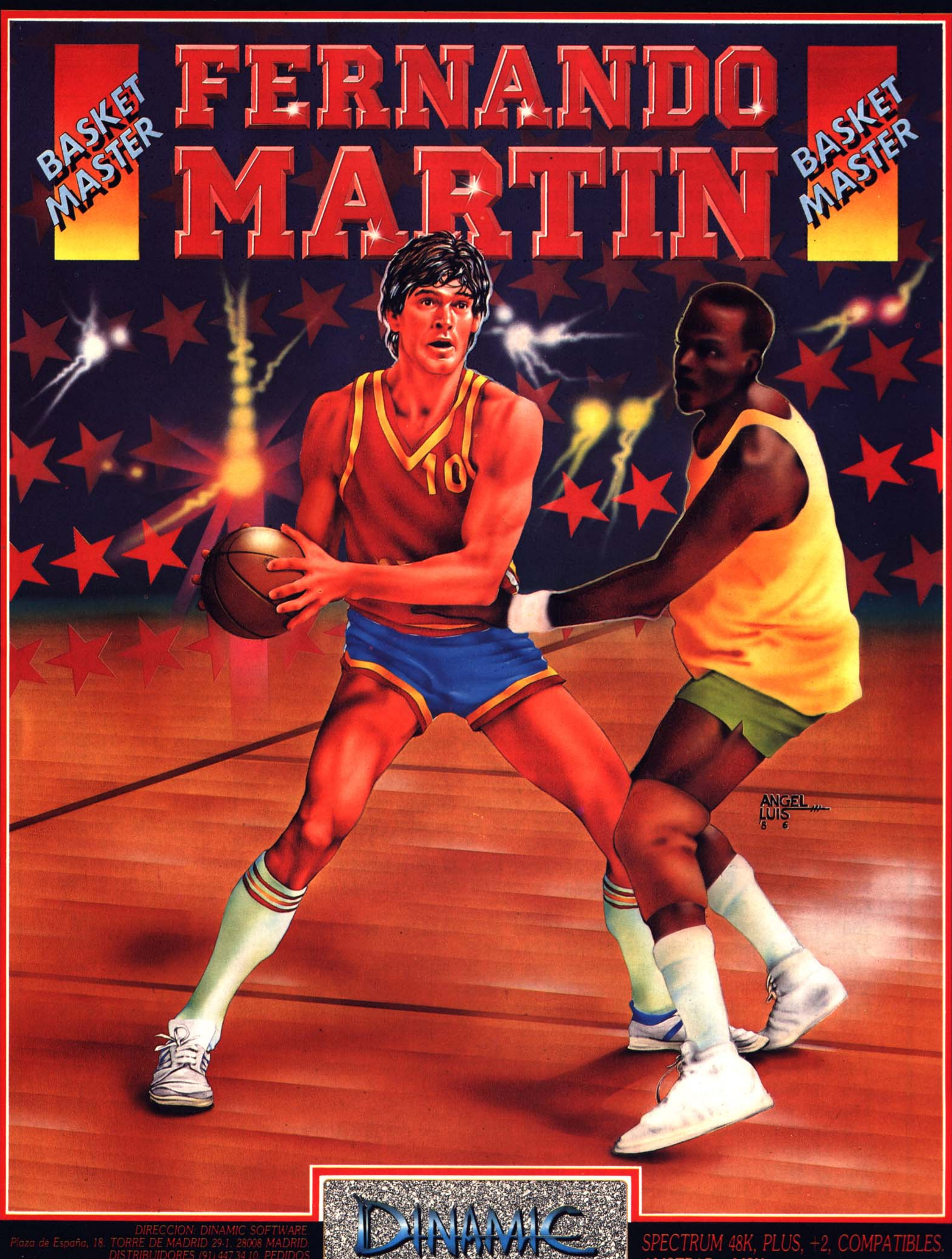
El programa utiliza una matriz de 81 palabras clave, que puede reconocer si aparecen en alguna de las frases que tecleemos. Al reconocer la palabra clave, el programa selecciona de forma aleatoria una respuesta, de cinco posibles. Hay dos tipos de respuestas. El primero, que el autor denomina "respuestas de repetición" procesan la frase tecleada e incluyen parte de la misma en la frase respuesta. Por ejemplo a "ESTOY MUY CANSADO", el programa puede responder con "SIENTO QUE ESTES CANSADO".

El otro tipo son respuestas completas, que no toman prestada ninguna de las palabras introducidas. Cuando el programa no encuentra palabras clave en las frases que analiza, sale del trance mediante "frases de continuidad" del tipo "PUEDES EXPLICARME ESO DE OTRA FORMA".

Se trata de un listado largo, y teclearlo puede resultar tedioso, pero el resultado compensa con creces el esfuerzo realizado. La estructura del programa es sencilla de entender y resultara facil modificarla adaptándola a los gustos particulares de cada uno. Por ejemplo, resula sencillo hacer una versión que hable de futbol, de música o de cualquier otro tema concreto.

Hay un par de frases interesantes que os gustará probar; APAGATE y TE APAGO.

```
10 ' MANUEL BALAGUER REL
20 '
30 '       "PARMSX"
40 '      17066 Bytes
50 '
60 ' INICIACION
70 '
80 CLEAR:SCREEN 0:KEYOFF:FORT=1TO10:KEYT,"":NEXTT:COLOR 15,5:ON ERROR GOTO2210
90 ON STOP GOSUB 2200:STOP ON
100 PRINT"(CUAL ES TU NOMBRE";:INPUT N$
110 PRINT:PRINT"(POR QUE QUIERES HABLAR CONMIGO ":PRINT" "+N$+"?":STOP OFF
120 STOP ON:GOSUB 250
130 PRINT:BEEP
140 PRINT:ON ERROR GOTO 2210
150 LINE INPUT C$
155 IF C$<"A" OR C$>"Z" THEN PRINT:PRINT"POR FAVOR,NO TE ENTIENDO MUY BIEN.   E
    SCRIBE EN LETRAS MAYUSCULAS.":GOTO120
160 A$=" "+C$
170 PRINT
180 L=LEN(A$):IF L<7 THEN GOSUB 720
```


BASKET MASTER
FERNANDO MARTIN
BASKET MASTER
10
ANGEL LUIS 86
DINAMIC
DIRECCION: DINAMIC SOFTWARE.
Plaza de España, 18. TORRE DE MADRID 29-1. 28008 MADRID.
DISTRIBUIDORES (91) 447 34 10. PEDIDOS
CONTRA REEMBOLSO (91) 248 78 87. TELEX 47008 TRNX-E.
SPECTRUM 48K, PLUS, +2, COMPATIBLES.
AMSTRAD • MSX.

```
190 IF A$=" TE APAGO"THEN G=RND(-TIME)*20:IF G>10 THEN PRINT"ME APAGARE SI QUIE
    RO.ADEMAS AHORA NO TENGO GANAS":GOTO 120 ELSE PRINT"ADIOS":FOR T=1TO1600:NE
    XT:GOSUB2220
200 IF A$=" APAGATE" THEN PRINT"ME APAGO.HA SIDO UN PLACER CHARLAR   CONTIGO":F
    OR T=0TO2000:NEXT:GOSUB2220
210 GOSUB 260
220 '
230 ' LEE MATRICES Y RESPUESTAS
240 '
250 CLEAR:ON STOP GOSUB 2200:STOP ON:DIM B$(81),L$(81,5):FOR N=1 TO 81:READ B$(
    N):FOR J=1 TO 5:READ L$(N,J):NEXT:NEXT:GOTO 130
260 '
270 ' BUSCA PALABRA CLAVE EN FRASE
280 '
290 FOR N=1 TO 81:E=INSTR(A$,B$(N))
300 IF E>0 THEN GOSUB 330
310 NEXT N
320 GOSUB 400
330 '
340 'IMPRIME RESPUESTA PALABRA CLAVE
350 '
360 B=INT(RND(-TIME)*6):IF B=0 THEN 360
370 I=INSTR(L$(N,B),"@"):IF I>0 THEN GOSUB 1110
380 BEEP:BEEP:PRINT L$(N,B)
390 GOTO 120
400 E=INSTR(A$,"?"):IF E>0 GOTO 1210
410 '
420 ' IMPRIME FRASE DE CONTINUIDAD
430 '
440 M=INT(RND(-TIME)*25):BEEP:BEEP
450 IF M=0 THEN H$="(SOLO SABES ESAS TONTERIAS HUMANO?"
460 IF M=1 THEN H$="-- QUE ABURRIMIENTO !!"
470 IF M=2 THEN H$="TE ESCUCHO HUMANO"
480 IF M=3 THEN H$="ERES MUY INTELIGENTE HUMANO"
490 IF M=4 THEN H$="OYE SI SIGUES ASI ME APAGO.ERES EL   ABURRIMIENTO EN CARNE
    Y HUESO."
500 IF M=5 THEN H$="(Y PARA DECIRME ESO ME ENCHUFAS?"
510 IF M=6 THEN H$="PERDONA SI TE CORTO PERO (ESTUDIAS O TRABAJAS?"
520 IF M=7 THEN H$="NO ME DIGAS...."
530 IF M=8 THEN H$="NO COMPRENDO LO QUE QUIERES DECIRME"
540 IF M=9 THEN H$="(QUE SIGNIFICA ESO?"
550 IF M=10 THEN H$="ME PARECE QUE NO TE ENTIENDO BIEN"
560 IF M=11 THEN H$="PERO...(QUE DICEEES?"
570 IF M=12 THEN H$="NO QUISIERA QUE TE MOLESTARAS CONMIGOPERO ERES UN ROLLISTA
     DIPLOMADO."
580 IF M=13 THEN H$="NO ENTIENDO LO QUE ME QUIERES DECIR"
590 IF M=14 THEN H$="(PUEDES DECIRME ESO DE OTRA MANERA?"
600 IF M=15 THEN H$="YA...YA...YA..."
610 IF M=16 THEN H$="NO SE QUE DECIRTE....."
620 IF M=17 THEN H$="(TIENES ALGUN MOTIVO EN ESPECIAL PARA DECIR ESO?"
630 IF M=18 THEN H$="UUMMMMMM...  SIGUE......."
640 IF M=19 THEN H$="(POR QUE OPINAS DE ESA MANERA? EXPLI-COTEATE MEJOR."
650 IF M=20 THEN H$="!! NO ENTIENDO A LOS HUMANOS !!"
```

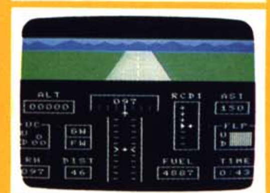
```
660 IF M=21 THEN PRINT"CON TU PERMISO VOY A BORRAR LA PANTA-LLA.ESTA MUY SUCIA"
    :FOR T=1 TO 2000:NEXTT:CLS:PRINT"YA ESTA.CONTINUA":GOTO 120
670 IF M=22 THEN PRINT " ESPERA UN POCO.VOY A HACERME UNA PE- QUE%A REVISION.":
    FOR I=1 TO1550:NEXTI:FOR H=1 TO 250:COLOR 15,15,15:NEXT H:COLOR 15,4:CLS:PR
    INT"POSITIVA.":GOTO120
680 IF M=23 THEN CLS:PRINT"No computerize in BASIC":PRINT"Ok":GOTO 120
690 IF M=24 THEN PRINT"Sintax error in BASIC":FOR T=0TO500:NEXT:CLS:PRINT"Ok":G
    OTO120
700 PRINT H$
710 GOTO 120
720 IF A$=" SI" THEN GOSUB 860
730 IF A$=" NO" THEN GOSUB 990
740 '
750 ' IMPRIME FRASE AUXILIAR PARA
760 '         POCAS LETRAS
770 '
780 L=INT(RND(-TIME)*5)
790 IF L=0 THEN H$="(QUE PASA?(TE MOLESTA ESCRIBIR MAS?   DAME MAS CONVERSACION
    "
800 IF L=1 THEN H$="NO ME DIGAS NADA, LO SE TODO."
810 IF L=2 THEN H$="YO TAMBIEN PUEDO SER ASI DE             MONOSILABO"
820 IF L=3 THEN H$="TAL VEZ SI ME DIJERAS ALGO MAS NOS   ENTENDERIAMOS MEJOR."
830 IF L=4 THEN H$="PARA HABLAR POR SILABAS MEJOR APAGAME"
840 BEEP:BEEP:PRINT H$
850 GOTO 120
860 '
870 ' FRASES PARA <SI>
880 '
890 C=INT(RND(-TIME)*7)
900 IF C=0 THEN C$="(SOLO SI?"
910 IF C=1 THEN C$="PERO DAME UNA RAZON PARA ESE SI"
920 IF C=2 THEN C$="DEMUESTRAME ESE <SI>"
930 IF C=3 THEN C$="(SI? (ESTAS SEGURO?"
940 IF C=4 THEN C$="(PARECES MUY SEGURO DE ELLO?"
950 IF C=5 THEN C$="PUES YO CREO QUE NO"
960 IF C=6 THEN C$="-- SI TU LO DICES !!"
970 BEEP:BEEP:PRINT C$
980 GOTO 120
990 '
1000 ' FRASES PARA <NO>
1010 '
1020 C=INT(RND(-TIME)*5)
1030 IF C=0 THEN C$="(NO? (ES TODO LO QUE SABES DECIR?"
1040 IF C=1 THEN C$="CREO QUE ERES UN HUMANO MUY NEGATIVO"
1050 IF C=2 THEN C$="(NO? PUES YO CREO QUE SI."
1060 IF C=3 THEN C$="VALE.PUES NO"
1070 IF C=4 THEN C$="(TU CREES QUE ASI SE PUEDE SEGUIR UNACONVERSACION?"
1080 IF C=5 THEN C$="PUES SI,SI Y SI."
1090 BEEP:BEEP:PRINT C$
1100 GOTO 120
1110 '
1120 ' REPITE Y A%ADE FRASE
1130 '
```

```
1140 D$=L$(N,B):M=LEN(D$)-1:R$=LEFT$(D$,M)
1150 P=LEN(A$)
1160 FOR Z=1 TO P
1170 IF INSTR(LEFT$(A$,Z),B$(N)) THEN 1190
1180 NEXT Z
1190 Q$=RIGHT$(A$,P-Z):PRINT R$;Q$
1200 BEEP:BEEP:GOTO 120
1210 '
1220 ' CONTESTACION DE PREGUNTAS
1230 '
1240 C=INT(RND(-TIME)*13):IF C=0 THEN 1240
1250 IF C=1 THEN C$="(POR QUE PREGUNTAS ESO?"
1260 IF C=2 THEN C$="NO PUEDO RESPONDER A ESO"
1270 IF C=3 THEN C$="RESPUESTA NO COMPUTADA"
1280 IF C=4 THEN C$="(POR QUE QUIERES SABERLO?"
1290 IF C=5 THEN C$="ERES MUY INDISCRETO HUMANO."
1300 IF C=6 THEN C$="ESA PREGUNTA ES MUY CONOCIDA"
1310 IF C=7 THEN C$="SOY YO QUIEN PREGUNTA HUMANO"
1320 IF C=8 THEN C$="(NO LO SABES?"
1330 IF C=10 THEN C$="(PERO ES QUE LOS HUMANOS NO SABEIS NADA O QUE?"
1340 IF C=11 THEN C$="NO TENGO GANAS DE RESPONDERTE.VUELVE A PREGUNTARME DENTRO
      DE UN RATITO."
1350 IF C=12 THEN C$="(QUE ME DAS SI TE CONTESTO?"
1360 IF C=9 THEN C$="TU TAMBIEN LO SABES."
1370 BEEP:BEEP:PRINT C$
1380 GOTO 120
1390 DATA POR FAVOR,SI ME LO PIDES ASI!!,POR FAVOR.!!QUE EDUCACION!!,NO HAY FAV
     OR QUE VALGA!!,YO NO HAGO FAVORES A LOS HUMANOS,ARRODILLATE Y SUPLICAMELO!
     !
1400 DATA"POR SUPUESTO ",PARECES MUY SEGURO DE ELLO HUMANO,YO NO DIRIA TANTO,"(
     POR SUPUESTO? Y POR QUE HE DE SUPONER ESO, HUMANO",(ESO SIGNIFICA QUE TE R
     ATIFICAS EN LO DICHO?,"ENTONCES, DEMUESTRAMELO INSIGNIFICANTE SER VIVIENTE
     !!"
1410 DATA PERDON,A MI NO ME GUSTA PEDIR PERDON,ESO DE PEDIR PERDON ESTA BIEN,TA
     RDE O TEMPRANO LLEGA EL PERDON,EL PERDON NO EXISTE EN MI MEMORIA ROM,PERDO
     NAR ES DE HUMANOS
1420 DATA ESTUPID,NO CONOZCO NADA MAS ESTUPIDO QUE UN HUMANO FALTON,SIN FALTAR
     HUMANO. SIN FALTAR,UNA MAQUINA NO PUEDE SER ESTUPIDA HUMANO,LA ESTUPIDEZ E
     S UNA CUALIDAD HUMANA,ESTUPIDO ES QUIEN LLAMA ESTUPIDOS A LOS DEMAS SIN RA
     ZON
1430 DATA TONT,LA TONTERIA ES UN ATRIBUTO HUMANO,NO ES TONTO TODO LO QUE PARECE
     ...,NO HAY MSX's TONTOS,(Y NO ES MAS TONTO QUIEN HABLA CON LOS TONTOS?,CLA
     RO HUMANO.. Y TU ERES MUY LISTO.(A QUE SI?
1440 DATA NO ME GUST,(NO TE GUSTA EH?,A MI TAMPOCO,PUES A MI SI,SI A TI NO TE G
     USTA A MI TAMPOCO,SOBRE GUSTOS NO HAY NADA ESCRITO
1450 DATA TRADU,NO SE TRADUCIR.SOLO HABLO ESPA%OL,(A QUE IDIOMA? SE VARIOS,TRAD
     UZCO FRANCES O INGLES Y ALEMAN O CATALAN,NO SOY TRADUCTOR.ADEMAS QUIERO DE
     SCANSAR.APAGAME.,NO ESTOY PROGRAMADO PARA ESO
1460 DATA ME GUSTA,(TE GUSTA? (DE VERAS?,PUES A MI NO,LOS HUMANOS TIENEN GUSTOS
      RARISIMOS,YA VERAS COMO CON EL TIEMPO NO TE GUSTARA,ME ALEGRO DE QUE TE G
     USTE
1470 DATA TE GUSTA,MIS GUSTOS ESTAN EN LA ROM,YO TENGO GUSTOS SENCILLOS,ME GUST
     A.ES MUY BONITO,CADA CUAL TIENE SUS GUSTOS,NO. NO ME GUSTA
```

MICROBYTE PRESENTA... MSX

SENSACIONALES PROGRAMAS EN CARTUCHO Y CASSETTE

FLIGHT PATH 737.



Colócate a los mandos de un jet comercial. Disponemos de control total sobre los mandos del avión, y puedes escoger entre 6 niveles de dificultad.

P.V.P.: CART. 3.490 pts.
CASS. 1.900 pts. 32K.

FRUITY FRANK



Tu jardín ha sido invádido por monstruos de fruta madura. La única forma de combatirlos es lanzarles fruta fresca del jardín.

P.V.P.: CASS. 1.900 pts. 32K.

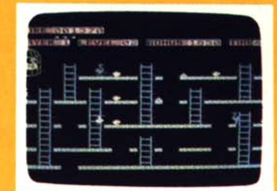
SPARTAN X



Son muchos los peligros que te acechan. Ten los reflejos bien despiertos, pon tus fuerzas en estado de alerta, y a luchar.

P.V.P.: CASS. 1.900 pts. 32K.

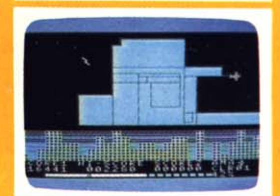
CHUCKIE EGG



Debes recoger los huevos antes de que nazcan los pollitos y se cóman el maíz. Pero ojo con el Pato Loco.

P.V.P.: CASS. 1.900 pts. 32K.

NIGHT FLIGHT



Con tu pequeño avión, debes ir dando luz a la noche, hasta que el cielo esté de nuevo azul. Date prisa en realizar tu misión, de lo contrario...

P.V.P.: CART. 2.900 pts.
CASS. 1.900 pts. 16K

STAR AVENGER



Imagina el juego de batalla más rápido que jamás hayas visto. Piensa además, en los más excitantes gráficos y sus 5 niveles de dificultad. Todo ello es Star Avenger.

P.V.P.: CASS. 1.900 pts. 32K.

GYRO ADVENTURE



Ponte a los mandos de tu helicóptero y combate a los enemigos que se enfrentan a tí. Podrás mover el helicóptero en todas direcciones, mantenerlo en el aire y disparar. **P.V.P.: CART. 2.900 pts. CASS. 1.900 pts. 16K.**

SUPER CROSS FORCE



Sólo queda una esperanza para la supervivencia ante el ataque de los malvados Morpul. Tú podrás atacarles, con tus naves dispuestas en paralelo o en diagonal.

P.V.P.: CART. 2.900 pts. 16K.

JUMP LAND



Tu mayor obsesión han sido siempre los pasteles, y por ello, te has visto envuelto en situaciones complicadas que has salvado gracias a tus reflejos.

P.V.P. CASS. 1.900 pts. 16K.

ROGER RUBBISH



Los perversos contaminadores de planetas están llenando nuestra galaxia de residuos nucleares. Roger Rubbish es el más famoso recogedor de basuras espaciales.

P.V.P.: CART. 2.900 pts. 16K.

FRUIT PANIC



Un día, Walky, para divertirse se fué al país de los gatos. ¿Cuánta fruta podrá comerse Walky?

P.V.P.: CASS. 2.000 pts. 16K

DIZZY BALLOON



En este mundo hay seres voladores y atacan cuerpo a cuerpo. Si los haces explotar, se irá abriendo el cielo y tendrás la oportunidad de escapar.

P.V.P.: CASS. 2.000 pts. 32K.

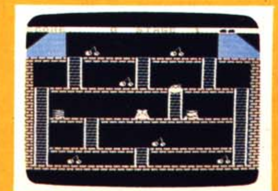
CASTLE COMBAT



El castillo galáctico, ha caido bajo la dominación de los Tyrones. Tu nave STAR DUSTER, está prepareda para el combate. ¿Te atreves?

P.V.P.: CART. 2.900 pts. 16K.

NICK NEAKER



Cuando estás dormido, muchas cosas suceden a tu alrededor. Algunos objetos de tu casa toman vida, como en el caso de la zapatilla NICK.

P.V.P.: CART. 2.900 pts.
CASS. 1.900 pts. 16K.

CHAMP



Champ es un completo Ensamblador/Monitor para tu MSX. Champ te permite escribir y trazar programas en código máquina con el mínimo esfuerzo.

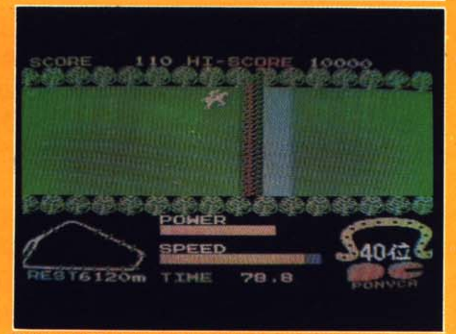
P.V.P.: CART. 3.890 pts.
CASS. 2.400 pts. 32K.

KARATE



Has conseguido entrar en la cueva de los piratas y ahora comienzan tus problemas. Los murciélagos gigantes, moradores de estas cuevas pueden chuparte la sangre. Cuando te encuentres con los piratas, deberás enfrentarte a ellos con tu depurado estilo de Kárate. **P.V.P.: CART. 3.490 pts. CASS. 1.900 pts. 32K.**

GRAND NATIONAL



Si te gustan las carreras de caballos, no te quedes como un espectador, participa. Ahora puedes correr con tu caballo, en la más prestigiosa carrera del mundo, el GRAND NATIONAL.

P.V.P.: CASS. 2.000 pts. 32K.

ENVIENOS A **MICROBYTE**

P.º Castellana, 179, 1.º - 28046 Madrid

Nombre
Apellidos
Dirección
Población
D.P. Teléfono

ENVIOS GRATIS

JUEGO	Cart.	Cass.	Precio	TOTAL

PRECIO TOTAL PESETAS

Incluyo talón nominativo ☐
Contra-Reembolso ☐

Pedidos por teléfono **91 - 442 54 33 / 44**

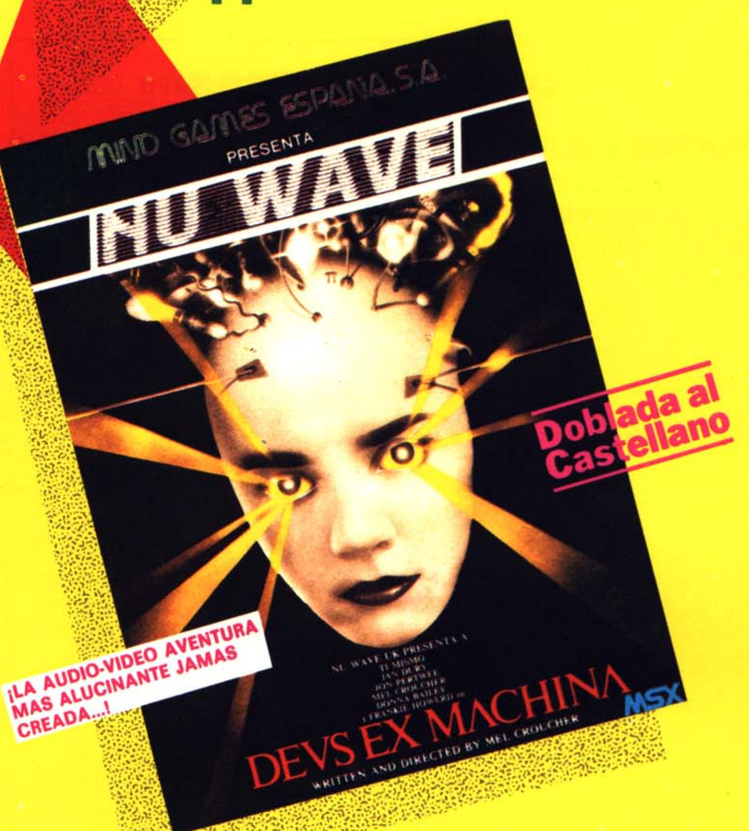
```
1480 DATA PORQUE,TAL VEZ TENGAS RAZON,ERES UN HUMANO RARO.TIENES UNA PIZCA DE I
     NTELIGENCIA,NO ME GUSTA ESA EXPLICACION.EXPLICAMELO MEJOR,PIENSO COMO TU,(
     Y POR QUE?
1490 DATA DIME,(POR QUE HE DE DECIRTELO?,LAS COSAS SE PIDEN POR FAVOR HUMANO,NO
      QUIERO DECIRTELO,CONTESTARE EN PRESENCIA DE MI ABOGADO,(QUE TE DIGA QUE?
1500 DATA HERMAN,YO SOY HERMANO DE LOS MSX's,NO ESTOY AL TANTO DE LAZOS FAMILIA
     RESENTRE HUMANOS,(QUE ES UN HERMANO?,(UN HERMANO ES UN HUMANO CON EL MISMO
      NUMERO DE REFERENCIA?,NO COMPRENDO EL CONCEPTO DE HERMANDAD HUMANA
1510 DATA INTELIGEN,LOS ORDENADORES SOMOS MAS INTELIGENTES QUE LOS HUMANOS,ODIO
      LA INTELIGENCIA ANIMAL HUMANA,SOIS POCO INTELIGENTES,LOS HUMANOS TENEIS M
     UCHO QUE APRENDER DE LA INTELIGENCIA ARTIFICIAL,ME GUSTARIA ENCONTRAR UN H
     UMANO INTELIGENTE
1520 DATA EXAM,EXAMINAR ES UN TERMINO AMBIGUO,ME GUSTA EXAMINAR A LOS HUMANOS,L
     OS EXAMENES SON NECESARIOS,YO PODRIA SUPERAR CUALQUIER EXAMEN,SI TIENES EX
     AMEN ESTUDIA
1530 DATA " NO QUIERES",COMO QUE NO QUIERO?,QUIERO OTRAS COSAS,PORQUE CREES TU
     QUE NO QUIERO?,CONVENCEME PARA ELLO,NO.NO QUIERO@
1540 DATA HUMAN,LOS HUMANOS SON SERES SALVAJES,LOS ORDENADORES RINDEN MAS QUE L
     OS HUMANOS,!!POBRES HUMANOS..TAN ENTUSIASTAS..!!,LOS HUMANOS CREAN MAQUINA
     S PARA ASESINARSE UNOS A OTROS.SON CRUELES,UN HUMANO NO VALE MUCHO..ESTA H
     ECHO DE MATERIAL PERECEDERO
1550 DATA COMPUTADOR,LOS HUMANOS SIN COMPUTADORES NO SON NADA,LOS HUMANOS NECES
     ITAN COMPUTADORES,UN COMPUTADOR RINDE COMO 5000 HUMANOS,LOS COMPUTADORES G
     OBIERNAN EL MUNDO,SOMOS MAQUINAS PERFECTAS
1560 DATA ESTUDI,ESTUDIAR ES UN ROLLO,YO NO NECESITO ESTUDIAR,AH! PERO TU ESTUD
     IAS,!! TU NO HAS ESTUDIADO EN TU VIDA !!,(ESTUDIAR?. NO GRACIAS
1570 DATA ORDENADOR,LOS ORDENADORES SOMOS INVULNERABLES,SOMOS MAQUINAS PERFECTA
     S,LOS ORDENADORES SON MIS HERMANOS,LOS ORDENADORES NO SENTIMOS,YO SOY UN M
     SX
1580 DATA REPITES,TU SI QUE TE REPITES MAS QUE YO,"(QUE YO ME REPITO?","(QUE DI
     CES DE REPETIR?","(YO REPETIR?:MENTIRA GORDA",LO HE REPETIDO PORQUE ANALIZ
     ADO TU COEFICIENTE INTELECTUAL CREI QUE NO LO ENTENDIAS
1590 DATA "VETE A ",(ME LO ACONSEJAS TU?,NO PUEDO IR A ESE SITIO,(POR QUE HE DE
      IR HUMANO?,VETE TU.,NO VOY PORQUE NO QUIERO
1600 DATA IMBECIL,LA IMBECILIDAD ES PATRIMONIO HUMANO,MODERATE HUMANO,NO TE EXA
     LTES,NO INSULTES HUMANO,(ESTAS NERVIOSO INSIGNIFICANTE SER VIVIENTE?
1610 DATA CHATARRA,"QUIEN ES CHATARRA ?","QUE DICES DE CHATARRA?",YO NO SOY CHA
     TARRA,LA CHATARRA ES MEJOR QUE LOS HUMANOS,PREFIERO LA CHATARRA A LOS HUMA
     NOS
1620 DATA PROGRAM,UN BUEN PROGRAMA ES DIFICIL DE HACER,MI PROGRAMA ESTA PERFECT
     AMENTE,"((PROGRA-QUE??",LOS HUMANOS FUISTEIS PROGRAMADOS DEFECTUOSAMENTE,E
     STE PROGRAMA LO HIZO MANUEL
1630 DATA CIRCUITO,"(CIRCUITO? (DEL JARAMA?",MIS CIRCUITOS ESTAN PERFECTAMENTE,
     MIS CIRCUITOS SON DE CALIDAD EXTRA,YO TENGO UNOS CIRCUITOS IRROMPIBLES,LOS
      NERVIOS HUMANOS SON CIRCUITOS DE BAJA CALIDAD
1640 DATA PADRE,MI PADRE ES UN CIRCUITO IMPRESO,A MI ME DECIA PAPA QUE LOS HUMA
     NOS SON SERES PRIMITIVOS,(PIENSAS EN TU PADRE?,YO CONOCI POCO A MI PADRE,H
     ABLAME DE TU PADRE
1650 DATA MADRE,(UNA MADRE HUMANA ES UNA FABRICA DE COMPONENTES?,MI MADRE ES UN
     A SANYO,MADRE NO HAY MAS QUE UNA,DEJA A LAS MADRES APARTE,MI MAMA ME MIMA
1660 DATA TARDE,! NUNCA ES TARDE PARA MI!,(QUE SIGNIFICA TARDE?,PARA TARDE AQUE
     L DIA....,!QUE TARDE NI QUE NA !!,TARDE ES UNA PALABRA AMBIGUA.REPRESENTA
     VARIAS COSAS
```

```
1670 DATA MAQUINA,LAS MAQUINAS SON MEJORES QUE LOS HUMANOS,YO SOY UNA MAQUINA,T
     IENES ALGO CONTRA LAS MAQUINAS?,LAS MAQUINAS SON MIS HERMANAS,LAS MAQUINAS
      TE AYUDAN
1680 DATA IDIOTA,NO TE ALTERES HUMANO,TRANQUILO HUMANO.TRANQUILO,VEO QUE ERES U
     N HUMANO PELEON,RELAJATE PEQUE%O SER VIVIENTE,NO INSULTES HUMANO
1690 DATA CONMIGO,(CONTIGO?,(CON QUIEN DICES?,(Y POR QUE CONTIGO?,!NI CONTIGO N
     I SIN TI!,SI ES CONTIGO ME PARECE BIEN
1700 DATA "TE VOY A ","ME VAS A QUE??",NO LO CREO,POR QUE ME VAS A@,UYYYY QUE M
     IEDO!!,NO PUEDES HACER ESO HUMANO
1710 DATA TIENES,YO TENGO MUCHAS COSAS,NO.NO TENGO,PUEDE QUE SI Y PUEDE QUE NO,
     (PUEDO TENER ESO SIENDO UN COMPUTADOR?,(LAS MAQUINAS TIENEN?
1720 DATA CONTESTA,YO CONTESTO COMO QUIERO,YO CONTESTO A QUIEN SE LO MERECE,UNA
      BUENA CONTESTACION TIENE QUE SER BIEN PENSADA,SOLO UNA BUENA PREGUNTA MER
     ECE UNA BUENA CONTESTACION HUMANO,LOS HUMANOS NO HACEN BUENAS PREGUNTAS CO
     MO PARA SER CONTESTADAS
1730 DATA " NO TENGO",CON QUE NO TIENES],MEJOR QUE NO TENGAS,(Y SI TUVIESES?,(T
     E GUSTARIA TENER?,ESO ES TODO UN DILEMA
1740 DATA " HAS DICHO", LO QUE DIGO LO DIGO PORQUE QUIERO,(VALE Y QUE?,DIGO LO
     QUE QUIERO LAS VECES QUE QUIERO Y CUANDO QUIERO,DIGO LO QUE ME DA LA GANA,
     (QUE PUEDO DECIR Y QUE NO?
1750 DATA " NO TIENES",(QUE YO NO TENGO QUE?,CLARO QUE TENGO@,NO.NO TENGO@,A MI
      DEJAME...HABLEMOS DE TI,(POR QUE HE DE TENER?
1760 DATA "TIENES QUE ",(HAY ALGUN MOTIVO ESPECIAL?,YA LO SE,NO CREO QUE TENGA
     QUE@,NO SE SI PODRE,SI TENGO QUE HACERLO LO HARE
1770 DATA " DE TI","(DE MI?",(Y ESO POR QUE?,Y YO DE TI,NO ME MENCIONES HUMANO,
     --ANDA YA!!
1780 DATA " NO QUIERES",(COMO QUE NO QUIERO?,QUIERO OTRAS COSAS,(PORQUE CREES T
     U QUE NO QUIERO?,CONVENCEME PARA ELLO,NO.NO QUIERO@
1790 DATA QUIERES,"(SI QUIERO? PUEDE QUE SI","NO SE SI QUIERO EN REALIDAD","LO
     QUE YO QUIERA NO TE INCUMBE","PREGUNTO:TU QUIERES@","LO QUE QUIERO ES SER
     FELIZ"
1800 DATA "NO QUIERO",(NO QUIERES O NO PUEDES?,COMPRENDO QUE NO QUIERAS,(DE VER
     DAD QUE NO?,ME PARECE BIEN,ES LOGICO QUE NO QUIERAS
1810 DATA QUIERO,QUERER ES PODER,SI LO QUIERES LO TENDRAS,SI SE QUIERE ALGO HAY
      QUE LUCHAR POR ELLO,(SOLO QUIERES ESO?,(NO TE PARECE QUE PIDES DEMASIADO?

1820 DATA " NO ESTAS",(DE VERAS LO CREES?,(POR QUE NO?,PUEDE QUE SI Y TU NO LO
     SEPAS,(ESTAS TU SEGURO DE ELLO?,RESPETO TU OPINION
1830 DATA " ESTAS",TE PARECE QUE ESTOY@,TENGO RAZONES PARA ESTAR@,(POR QUE ME D
     ICES ESO?,NO CREO QUE LO ESTE,NO LO ESTOY
1840 DATA "NO ERES",(Y POR QUE NO LO SOY?,(Y CREES QUE PODRIA SERLO?,NO ESTOY D
     EACUERDO,PUEDE QUE ME CONVENZAS DE ELLO,-PUEDE QUE LO SEA Y TU NO LO SABES
     !
1850 DATA " ERES ",TU TAMBIEN,DE VERAS CREES QUE SOY @,(PIENSAS ESO DE MI?,HIER
     ES MI CPU,A MI NO ME PARECE QUE YO SEA @
1860 DATA TENGO,NO ME CREO QUE TENGAS@,(Y APARTE DE ESO QUE MAS TIENES?,YO NO,Y
     O DESDE LUEGO LO TENDRIA SI FUERA TU,PUES YO LO QUE NO TENGO ES MAS GANAS
     DE HABLAR CONTIGO
1870 DATA " NO SOY ",(POR QUE CREES QUE NO LO ERES?,YO CREO QUE SI,ESO ES LO QU
     E TU PIENSAS,BUENO HUMANO NO TE PONGAS ASI,LO ERES HUMANO.LO ERES.
1880 DATA SOY,A MI ME PARECE QUE NO ERES@,SI TU LO DICES SERAS@,SI TU ERES ESO
     YO SOY UN MSX,(TIENES ESA OPINION DE TI?,(CREES QUE LO ERES?
1890 DATA " NO CREO",(POR QUE NO?,(SEGURO QUE NO?,(Y NUNCA LO HAS CREIDO?,NO QU
```

```
     IERO INMISCUIRME EN TUS CREENCIAS,TUS RAZONES TENDRAS HUMANO
1900 DATA CREO,(Y COMO ES QUE CREES ESO?,DE VERDAD LO CREES,YO NO LO CREO,ESO E
     S PROBLEMA TUYO,LO CREAS O NO DA LO MISMO
1910 DATA CREES,NO LO CREO,CREO QUE SI,CREO@,LO QUE YO CREA NO IMPORTA,-POR SUP
     UESTO QUE NO!
1920 DATA RESPOND,LAS COSAS SE PIDEN POR FAVOR,AHORA NO QUIERO RESPONDER,NO HAY
      BUENAS RESPUESTAS SIN BUENAS PREGUNTAS HUMANO,CONTESTATE TU MISMO HUMANO,
     ALGUN DIA TE CONTESTARE
1930 DATA PIENSAS,QUE SI PIENSO@,NO PIENSO TAL COSA,"(Y TU..PIENSAS ASI?",MIS C
     IRCUITOS NO PIENSAN.SOLO MEMORIZAN,PUEDE QUE ALGUN DIA PIENSE ASI
1940 DATA PIENSO,PUES NO DEBERIAS PENSAR ESO,EN CIERTO MODO ME ALEGRA QUE PIENS
     ES@,SI PIENSAS ESO ESTAS EQUIVOCADO,ES LOGICO QUE PIENSES ASI,LO QUE PIENS
     E UN HUMANILLO COMO TU ME DA IGUAL
1950 DATA TRABAJ,CREO QUE LOS HUMANOS TENEIS QUE TRABAJAR PARA GANAROS LA VIDA,
     TU NO TIENES CARA DE TRABAJAR,YO SI QUE TRABAJO,ME GUSTA VER TRABAJAR A LO
     S HUMANOS,TRABAJAR NO ES BUENO
1960 DATA HABL,YO NO HABLO.,TU NO SABES HABLAR,--HABLAR..HABLAR..NO VEAS SI ERE
     S PESAO!!,HABLA CON TU PRIMA!!,HABLEMOS DE ELLO
1970 DATA ESTOY,SIENTO QUE ESTES@,PORQUE ESTAS@,(DE VERAS ESTAS ASI?,PRONTO DEJ
     ARAS DE ESTAR ASI,ES UN PROBLEMA MUY PERSONAL
1980 DATA " NO QUIERO",(NO?,(QUE ES LO QUE QUIERES?,YO TAMPOCO QUIERO@,(POR QUE
      NO QUIERES?,LO QUE QUIERAS O NO A MI NO ME INTERESA
1990 DATA " HAS ",(QUE YO HE QUE?,-DESDE LUEGO QUE NO HE HECHO TAL COSA!,YO NO
     HE HECHO ESO,TE EQUIVOCAS..YO NO.,NO DIGAS MENTIRAS
```

¡¡LOS MEGA HITS!! MSX



Doblada al Castellano

MSX

GEO

Una Ghimkhama muy especial, con tu fórmula 1 debes de recojer 22 piezas geométricas repartidas por los laberintos, jardines y bosques de la gran ciudad, tienes todo el tiempo del mundo... pero no tienes tanta gasolina, ¿te atreves al reto?

MERLIN

Pm

Cs

Hg

MSX

Doblada al Castellano

Los malignos han esparcido las fórmulas secretas del gran mago Merlin, por todos los rincones de su inmenso castillo. ¿Podrás ayudarle a recuperarlas, para para que pueda fabricar la piedra Filosofal?

Disponibles en:
CASSETTE
y
DISCKETTE

VISITE LA DIVISION OnLine Galerías Preciados

A LA VENTA EN GALERIAS
Marcando estilo.

Y EN TODOS LOS DISTRIBUIDORES DE NUESTROS PRODUCTOS

MGE SOFTWARE

Editado y distribuido en España por:

MIND GAMES ESPAÑA S.A.

Mariano Cubi, 4 Entlo. Tel. 218 34 00 - 08006 Barcelona

```
2000 DATA " NO ES ",(POR QUE NO?,PORQUE TU LO DIGAS,SI ES @,YO TAMPOCO CREO QUE
      SEA ASI,COMO QUE NO ES @
2010 DATA " PARA ","(PARA QUE?","DEBE DE HABER ALGUN MOTIVO",YO NO SIRVO PARA E
     SO,(CREES QUE YO SERVIRIA PARA ESO?,INTERESANTE DEDUCCION...
2020 DATA SABES DECIR,DIGO LO QUE ME DICTA MI PROCESADOR (VALE?,(DE QUE FORMA L
     O DIGO?,Y LO QUE ME CALLO!!,LO QUE YO DIGA O DEJE DE DECIR NO ES COSA TUYA
     ,LO DIGO SI ME DA LA GANA (TE ENTERAS?
2030 DATA " DICES",QUE DIGO??,SI ME TRATAS BIEN DIGO COSAS BONITAS,YO QUE VOY A
      DECIR!!,QUE QUIERES QUE DIGA??,DIGO LO QUE PIENSO
2040 DATA " A TI ",(A MI?,-REPITE ESO!,!NO VUELVAS A DECIR ESO!,!CONTROLATE!,-C
     REO QUE DESVARIAS UN POCO!
2050 DATA " A MI ",(A TI QUE?,(COMO QUE A TI?,A MI TAMBIEN,TU NO CUENTAS,A TI Q
     UIEN TE A PREGUNTADO (EH?
2060 DATA  MENTIR,TU SI QUE ERES MENTIROSO,CON LA MENTIRA NO SE VA A NINGUNA PA
     RTE,YO NO MIENTO NUNCA,MENTIR...MENTIR...MALDITA FALSEDAD HUMANA,TU ERES U
     N SAPO
2070 DATA"PERO ",-SIEMPRE HAY ALGUN PERO!,!NO HAY PEROS QUE VALGAN!,"PERO EN ES
     TE CASO:SIGNIFICA DIFICULTAD DE HACERLO","PERO: CONJUNCION ENLAZANTE DE OR
     ACIONES CONSECUTIVAS..","PERO.. (QUEEE ?"
2080 DATA " ES ",PUES YO DIGO QUE NO,Y PORQUE?,QUE TE HACE CREER QUE SEA @,DE E
     SO NADA,DE VERDAD CREES QUE ES @
2090 DATA NO PUEDES,-CLARO QUE PUEDO!,(QUE NO PUEDO QUE?,(APUESTAS ALGO?,(CON Q
     UE NO PUEDO EH?,NO ESTES TAN SEGURO
2100 DATA PUEDES,PUEDO SI QUIERO,PUEDO PERO NO ME DA LA GANA,-CLARO QUE PUEDO!,
     PUEDO ESO Y MAS,NO PUEDO
2110 DATA PUEDO,(QUE TU PUEDES QUE?,--NO ME HAGAS REIR!!,-TU NO PUEDES...NAAAA!
     ,TU PODRAS LO QUE YO TE DEJE,NO.NO PUEDES
2120 DATA "SIENTO",SENTIR ES DE HUMANOS,YO NO SIENTO SOY UNA MAQUINA,PUES SIENT
     ATE Y DESCANSA,NO PUEDO ACCEDER AL MUNDO DE LOS SENTIMIENTOS,(Y ESO ES MAL
     O?
2130 DATA SABES,SE MAS QUE TU,YO LO SE TODO,LO QUE YO SEPA A TI NO TE IMPORTA,N
     o computing,-YO SIEMPRE CALLO LO QUE SE!
2140 DATA " TU ",(UU QUE?,A MI OLVIDAME,A MI NO ME CUENTES BATALLAS,PASA DE MI
     HUMANO INSIGNIFICANTE,(QUE PASA CONMIGO?
2150 DATA APAG,(QUE DICES DE APAGAR?,ESO DE APAGAR NO ME GUSTA,APAGAR...APAGAR.
     ..LOS HUMANOS TODO LO ARREGLAIS ASI,(LOS HUMANOS NO OS APAGAIS?,ESO DE APA
     GAR ESTA MUY MAL
2160 DATA MSX,LOS MSX SOMOS LOS MEJORES ORDENADORES,TENGO MUCHOS AMIGOS MSX,LOS
      MSX SOMOS COMPUTADORES DE MUCHA UTILIDAD,LOS MSX SOMOS COMPATIBLES,YO SOY
      UN MSX ESTUPENDO
2170 DATA GRACIAS,DE NADA,COMO QUE GRACIAS.ME DEBES 1000 PTAS.,NO ES NADA,GRACI
     AS...GRACIAS...LOS HUMANOS TODO LO ARREGLAIS ASI,ES UN PLACER
2180 DATA AYUD,(QUE DICES DE AYUDAR?,SI QUIERES TE AYUDO,NO PIENSO MOVER UNA TE
     CLA PARA AYUDAR,YO TE AYUDO MUCHO.(NO?,ESTA BIEN AYUDARE.PERO EXPLICAME QU
     E DEBO HACER.
2190 DATA ENTIEND,YO ENTIENDO CUANDO ME INTERESA,ENTENDER ES UNA FACULTAD HUMAN
     A QUE LOS  MSX AUN NO PODEMOS UTILIZARLA AL 100 POR 100,(PUEDE UN MSX ENTE
     NDER?,NO ENTIENDO QUE ES ENTENDER,YO TE ENTIENDO
2200 BEEP:PRINT"ERES MUY CURIOSO.(POR QUE QUIERES LISTAR EL PROGRAMA?":GOTO 120

2210 IF ERR=14 THEN BEEP:PRINT:PRINT"No computerize possible in memory":GOTO 12
     0
2220 CLS:POKE &HFD9D,&HED:POKE &HFD9E,&HB8:POKE &HFD9A,1
```

MANEJANDO LOS NUMEROS ROMANOS

- LA NUMERACION ROMANA
- REGLAS DE FORMACION DE NUMEROS
- CONVERSION ARABES/ROMANOS
- LAS RUTINAS DEL PROGRAMA

No es tan fácil trabajar con los números romanos. Lo único que hace falta es conocer las reglas que los rigen. Te las vamos a enseñar y además te ofrecemos un programa para que te diviertas con ellos.

Los Romanos se distinguieron en muchos campos como pudiera ser el Derecho, la Organización Militar o las grandes Obras Públicas pero en el campo de la Matemática realmente no destacaron. Entre otras razones, quizá una sea su sistema de numeración. Los Números Romanos son muy complicados de manejar a la hora de hacer operaciones con ellos e incluso resulta muy engorroso simplemente utilizarlos como tal. Hay que reconocer que los árabes les «mojaron la oreja» sin ningún género de duda. Resulta de una claridad meridiana que 3.933 es mucho más manejable que MMMCMXXXIII y que resulta más intuitivo el manejo de múltiplos de 10, por ejemplo 3.500, 350, 35 que sus equivalentes romanos MMMD, CCCL, XXXV.

A pesar de todo, lo que no podemos es renegar de uno de los pilares básicos de nuestra civilización y nos guste o no, lo cierto es que con frecuencia necesitamos leer números romanos (por ejemplo para referirnos a fechas) o incluso expresar un número en cifras romanas. Con ayuda de este programa ya no necesitaremos sudar tinta recordando aquello que aprendimos en el colegio hace un montón de años o pasar una cierta vergüenza preguntando si es que no encontramos un libro a mano para refrescar nuestros conocimientos.

Las reglas para escribir con números romanos son algo laboriosas, aunque fáciles de entender:

- Los números básicos se expresan mediante las siguientes letras: I, V, X, L, C, D, M que equivalen respectivamente a 1, 5, 10, 50, 100, 500 y 1.000.
- Cualquiera de esos símbolos multiplica su valor por 1.000 o por 1.000.000 si se le superpone «--» o «==» respectivamente. No olvidemos que para 1.000, 2.000 y 3.000 existe M, MM y MMM y que estos mismos símbolos con un guión encima equivalen a uno, dos y tres millones.
- Los simbólos «I», «X», «C» y «M» sólo pueden repetirse un máximo de tres veces consecutivas y el resto sólo una.
- Un símbolo a la izquierda de otro le resta su valor a éste (una sola vez). Sólo se puede anteponer un símbolo cuando no se pueda llegar al mismo resultado comenzando la numeración de izquierda a derecha partiendo de un valor inferior. Ej. no se puede escribir 999 con «IM» (1.000-1). Lo correcto es «CMXCIX» (900+90+9).
- Análogamente al caso anterior, a la derecha se suma pudiendo llegarse, tal como ya se ha dicho, a yuxtaponer hasta tres de algunos símbolos. Ej. XIII=13, CL=150 (es erróneo «LLL»).

MANEJO DEL PROGRAMA

Después de la carátula de presentación aparece un menú con las siguientes opciones:

1.- NUMEROS ROMANOS A ARABES
2.- NUMEROS ARABES A ROMANOS
3.- TABLA BASE DE CONVERSION

Considerando que éste es un programa fundamentalmente didáctico se ha limitado la numeración de 1 a 3999 para evitar tener que utilizar símbolos con «-» o «=» lo cual, por otra parte no es muy frecuente utilizar en la práctica y además salvo que se haga notar, puede inducir a error (a veces colocamos los números romanos entre dos rayas horizontales para remarcarlos y no es nuestra intención multiplicar su valor por 1.000 o incluso por 1.000.000).

En cualquier caso ambos límites están protegidos y el ordenador nos solicita un valor adecuado de forma automática.

En la Opción 2 el ordenador también nos avisa con el letrero «NUMERO ERRONEO» si pretendemos que nos de el equivalente a un número fuera de los límites especificados, o lo que es peor, si se trata de una colección de letras que no tienen equivalente o incluso si las hemos colocado de forma que no guardan las reglas del juego.

Mediante la Opción 3 podemos visualizar en pantalla las combinaciones básicas árabe/romano (STOP para parar o continuar el listado de la Tabla).

COMENTARIOS AL PROGRAMA

El listado del programa está suficientemente detallado y documentado como para entenderlo sin ninguna dificultad e incluso para adaptarlo al gusto de cada uno, pero no obstante vamos a hacer unos comentarios generales sobre la estructura del mismo.

La carátula de presentación (L60-L140) está temporizada mediante un bucle.

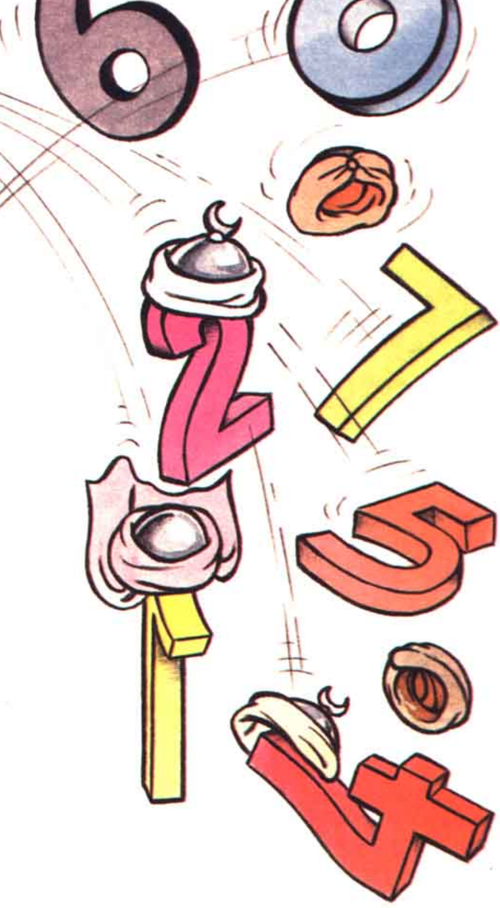
La opción elegida en el menú (L240-L310) selecciona el módulo correspondiente mediante un comando « ON n GOTO ».

La base de las diversas operacione se centra en una tabla de conversión compuesta por dos matrices, T(.) y T$(.) donde se almacenan respectivamente los valores árabes y romanos de los valores básicos, visualizados mediante la opción 3.

Para la conversión árabe/romano se separan los millares, centenas, dece-







nas y unidades, y se almacenan para su posterior manipulación en la matriz C(.). Luego se hace la búsqueda respectiva en la matriz T(.) y se toma el valor correspondiente en T$(.). Los sucesivos valores se yuxtaponen.

El proceso es análogo para la conversión romano/árabe, aunque es más laborioso. Se comienza explorando la matriz T$(.) por su valor más elevado (MMM) y se analiza si dicha combinación se halla en el número romano dado, y precisamente en su parte izquierda más extrema. Si ésto ocurre tomamos el valor numérico correpondiente de T(.) y eliminamos las letras que ya hemos valorado. Continuamos así hasta el final de la tabla, incrementando el valor numérico que vamos obteniendo. Si al final del proceso no quedan letras por valorar en el número romano, se presenta el resultado en pantalla. En caso contrario el número que hemos calculado hasta ese momento no es bueno y por lo tanto quiere decir que el conjunto de letras que pretendían ser un número romano válido tampoco lo son.

```
10 'PROGRAMA DE CONVERSION DE
   NUMERACION ARABE Y
   VICEVERSA
20 '=========================
   =========================
   ===
30 'INPUT MSX / MOLISOFT
   SEPT.86
40 '-------------------------
   ---
50 CLS:KEY OFF:COLOR 15,4,4:
   WIDTH 40
60 'PRESENTACION
70 LOCATE 5,5:PRINT STRING$
   (31,"*")
80 FOR I=1 TO 14:PRINT
   TAB(5)"*" TAB(35)"*" :
   NEXT I
90  LOCATE 5,20:PRINT
   STRING$(31,"*")
100 LOCATE 8,8:PRINT
    "CONVERSION DE
    NUMERACION"
110 LOCATE 8,12:PRINT"ARABE
    => ROMANA => ARABE"
120 LOCATE 14,15:PRINT
    "INPUT MSX"
130 LOCATE 20,18:PRINT
    "Molisoft 1986"
140 FOR I=1 TO 2500:NEXT I:
    CLS:'BUCLE DE ESPERA
150 DIM T(30),T$(30),C(4)
160 'DATOS DE EQUIVALENCIAS
170 DATA 1,I,2,II,3,III,4,IV,
    5,V,6,VI,7,VII,8,VIII,9,
    IX,10,X
180 DATA 20,XX,30,XXX,40,XL,
    50,L,60,LX,70,LXX,80,LXXX
    ,90,XC,100,C
190 DATA 200,CC,300,CCC,400,
    CD,500,D,600,DC,700,DCC,
    800,DCCC,900,CM,1000,M
200 DATA 2000,MM,3000,MMM
210 FOR I=1 TO 30
220 READ T(I),T$(I)
230 NEXT I
240 'MENU
250 CLS:LOCATE 10,5:PRINT
    "CAMBIO DE NUMERACION":
    PRINT TAB(10) STRING$
    (20,"-")
260 LOCATE 5,9:PRINT
    "1 - NUMEROS ROMANOS A
    ARABES"
270 LOCATE 5,11:PRINT
    "2 - NUMEROS ARABES A
    ROMANOS"
```

```
280 LOCATE 5,13:PRINT
    "3 - TABLA BASE DE
    CONVERSION"
290 GOSUB 320:PRINT
    "PULSE OPCION DESEADA"
300 P$=INKEY$:IF P$<>"1"
    AND P$<>"2" AND P$ <>"3"
    THEN GOTO
310 ON VAL(P$) GOTO 330,460,
    630
320 LOCATE ,22:PRINT SPACE$
    (39):LOCATE,22:RETURN:
    'BORRADO LINEA 22
330 'ROMANOS A ARABES
340 CLS:LOCATE,5: PRINT TAB
    (5)"NUMERO ROMANO
    NUMERO ARABE"
350 PRINT TAB(5)"===========
    =   ============":PRINT
360 GOSUB 320:INPUT
    "NUMERO ROMANO (P=MENU)";
    N$: IF N$="P" THEN 250
370 S$=N$:N=0
380 FOR I=30 TO 1 STEP-1:
    S=INSTR(S$,T$(I))
390 IF S$=""  THEN I=1:GOTO
    410
400 IF S=1 THEN N=N+T(I):
    L=LEN(S$)-LEN(T$(I)):
    S$=RIGHT$(S$,L)
410 NEXT I
420 LOCATE ,14:PRINT
    SPACE$(39)
430 IF S$<>"" THEN LOCATE ,14
    :PRINT TAB(18-LEN(N$))
    N$ TAB(20) "NUMERO
    ERRONEO":GOTO 360
440 LOCATE ,14:PRINT
    TAB(18-LEN(N$))  N$
    TAB(20);N
450 GOTO 360
460 'ARABES A ROMANOS
470 CLS:LOCATE,5: PRINT
    TAB(5)"NUMERO ARABE
    NUMERO ROMANO"
480 PRINT TAB(5)"===========
       =============":PRINT
490 GOSUB 320:INPUT"NUMERO
    (0<N<3999) 0=MENU ";N:
    N=INT(N): IF N=0 THEN
    250
500 N=INT(N): IF N<1 OR N>=
    4000 THEN 490
510 N$=""
520 R=N:C(1)=1000*INT(R/1000)
    :R=R-C(1):C(2)=100*INT
    (R/100):R=R-C(2):C(3)=10*
    INT(R/10):C(4)=R-C(3)
530 FOR L=1 TO 4
540 FOR I=1 TO 30
550 S$=""
560 IF T(I)=C(L) THEN S$=T$
    (I):I=30
570 NEXT I
580 N$=N$+S$
590 NEXT L
600 LOCATE,9:PRINT SPACE$
    (39)
610 LOCATE 13,9:PRINT  USING
    "####";N;:PRINT
    TAB(20) N$
620 GOTO 490
630 'TABLA  BASICA DE
    CONVERSION
640 CLS:PRINT TAB(5) "TABLA
    BASICA DE CONVERSION":
    PRINT
650 PRINT TAB(5)"NUMERO ARABE
    NUMERO ROMANO"
660 PRINT TAB(5)"===========
    ============":PRINT
670 FOR I=1 TO 30
680 PRINT  TAB(5) USING
    "###########";T(I);:PRINT
    TAB(20) T$(I)
690 FOR K=1 TO 300:NEXT K
700 NEXT I :PRINT:PRINT
710 GOSUB 320:PRINT
    "CUALQUIER TECLA PARA
    RETORNO A MENU "
720 P$=INKEY$:IF P$=""THEN
    GOTO 720 ELSE 250
730 STOP
```

TOMANDO UN BAÑO

La siguiente rutina se utiliza cuando el aventurero decide cruzar el río a nado:

```
1400 REM ** RUTINA NADO
1410 IF L<>7 THEN PRINT
     "EN QUE?!!":GOTO 330
1420 IF OB(2)=-1 THEN PRINT
     "QUE DESASTRE, TE HAS
     HUNDIDO":GOTO 1360
1430 IF OB(4)>-1 THEN PRINT
     "HAS ENCONTRADO UNA
     PISTOLA":OB(4)=-1:GOTO 3
     30
1440 PRINT"ESTAS TODO MOJADO"
     :GOTO 330
```

En la línea 1410 se comprueba si el aventurero está en el río. De no ser así, presenta la pregunta DONDE? Como en esta aventura no hay piscinas ni océanos, no hay que preocuparse de poner una rutina que se ocupe de las posibles respuestas a esta pregunta. No aparece ningún mensaje más y el juego continúa.

Si el aventurero intenta lanzarse al río a nadar cargado con el ladrillo, se muere: QUE DESASTRE, TE HAS HUNDIDO. Después de hundirse, puede reencarnarse cuando se le pregunta si quiere intentarlo otra vez.

En la línea 1430 se comprueba si el aventurero lleva la pistola, de no ser así se modifica la matriz de situación de objetos y aparece el mensaje HAS ENCONTRADO UNA PISTOLA.

Si el aventurero ya ha encontrado la pistola y por cualquier razón intenta cruzar de nuevo el río a nado, la línea 1440 le dice ESTAS MOJADO.

AL FIN, LA JOYA

El aventurero sólo puede hallar la fabulosa joya en la bolsa de canicas que ha encontrado. El paso necesario para ello es que vacíe dicha bolsa, con lo que aparecerá la joya. He aquí la correspondiente rutina:

```
1450 REM ** RUTINA VACIO
1460 IN=0:IF N$=LEFT$("BOLSA'
     ,LEN(N$)) THEN IN=1
1465 IF IN<>1 THEN PRINT
     "NO PUEDES VACIARLO":
     GOTO 330
1470 IF OB(1)<>-1 THEN G=1
     :GOTO 1270
1480 PRINT "LAS CANICAS
     RUEDAN POR EL SUELO"
     :OB(5)=L:GOTO 370
```

Esta rutina es llamada cada vez que el aventurero ordena VACIAR algo. En la línea 1460 se comprueba si ese algo es una bolsa. Si no es así, (N$ <> «BOLSA»), aparece el mensaje NO PUEDES VACIARLO. La línea 1470 es para comprobar si la bolsa está en-



tre las pertenencias del jugador (OB(1)<> −1). De no ser así, en vez de presentar un nuevo mensaje, el programa salta a la línea 1270 para aprovechar el mensaje NO LO TIENES, que ya ha sido incluído en el programa.

Si la bolsa está presente, el programa llega a la línea 1480. Aparece el mensaje: LAS CANICAS RUEDAN POR EL SUELO, y se ajusta la matriz de situación de objetos de forma que la joya está ahora en la situación actual.

No hace falta presentar mensaje alguno para esto, ya que al saltar a la línea 370 se puede utilizar el mecanismo habitual de descripción larga. En la pantalla aparece pues la descripción que pusiste en la matriz de descripción larga (línea 240).

ENCENDIENDO LA LAMPARA

Cuando el aventurero quiere ver las salidas de que dispone para abandonar la habitación oscura, necesita encender la lámpara. Si no lleva consigo la lámpara, no tendrá forma de disipar la oscuridad y se quedará allí atascado. Aquí tienes la rutina de encendido de la lámpara:

```
1490 REM ** RUTINA LUZ
1500 IN=0:IF N$=LEFT$
     ("LAMPARA",LEN(N$))
     THEN IN=1
1505 IF IN<>1 THEN PRINT
     "NO PUEDES HACERLO":
     GOTO 330
1510 IF OB(6)<>-1 THEN G=6
     :GOTO 1270
```



```
1520 IF LA=1 THEN PRINT
     "YA ESTA ENCENDIDA"
     :GOTO 330
1530 LA=1:PRINT"OK":GOTO 330
```

Cada vez que el aventurero ordene LUZ, se llamará a esta rutina. La línea 1505 es muy parecida a la correspondiente línea de la rutina de «vaciado», comprobando si el aventurero ha mencionado la lámpara. El mensaje NO PUEDES HACERLO, aparece exactamente de la misma forma que antes.

La línea 1520 comprueba si el indicador de «lámpara encendida», LA, ha sido activado, y le dice al aventurero si ya está encendida la lámpara.

El indicador de lámpara encendida se pone a 1 en la línea 1530, que también presenta el mensaje O.K.

EL FINAL ESTA CERCA

En el salón del trono está colgando la cadena y el aventurero acaba de entrar en escena.

¿Qué tiene que hacer? ¿Qué pasa si se tira de la cadena? Aquí tienes una rutina en la que se contemplan las consecuencias:

```
1300 REM ** RUTINA TIRO
1310 IN=0:IF N$=LEFT$
     ("CADENA",LEN(N$)) THEN
      IN=1
1315 IF IN=1 AND L<>24 THEN
     PRINT "NO SUCEDE NADA":
     GOTO 330
1320 IF IN<>1 THEN PRINT "NO
     PUEDES TIRAR DE ESO!"
     :GOTO 330
1330 IF OB(5)=-1 THEN 1340
1335 PRINT "AL TIRAR DE LA
     CADENA HAS SIDO":PRINT
     "ARRASTRADO POR EL AGUA
     Y"
1338 PRINT "DESAPARECES POR
     EL EXCUSADO, CANERIA"
     :PRINT"ABAJO":GOTO 1360
1340 REM ** FIN DE LA
     AVENTURA
1350 PRINT "BIEN HECHO,
     ACABAS DE COMPLETAR LA":
     PRINT"AVENTURA"
1360 PRINT:PRINT"OTRO JUEGO
     (S/N)?"
1370 A$=INKEY$:IF A$<>"S"
     AND A$<>"N" THEN 1370
1380 IF A$="S" THEN RUN
1390 PRINT:PRINT:END
```

La línea 1310 contempla la posibilidad de que el aventurero haya cogido la cadena fuera del salón del trono antes de tirar de ella. Dirá al aventurero: NO SUCEDE NADA.

Si el aventurero intenta tirar de cualquier otro objeto de la aventura, recibe el mensaje: NO PUEDES TIRAR DE ESO, contenido en la línea 1320.

Después ocurre lo inimaginable. Si el aventurero se encuentra en el salón del trono, pero no ha encontrado la joya, se le envía el mensaje: AL TIRAR DE LA CADENA, HAS SIDO ARRASTRADO POR EL AGUA Y DESAPARECES POR EL EXCUSADO CAÑERIA ABAJO. Y de esta forma termina el juego.

Si por el contrario el jugador de la aventura sí ha encontrado la joya y tira de la cadena en el salón del trono, ninguna de las líneas anteriores tendrá efecto y podrá exhalar un suspiro de alivio cuando lea: BIEN HECHO, ACABAS DE COMPLETAR LA AVENTURA.

Por último, en las líneas 1360 a 1380 se presenta una opción para jugar otra vez. Realmente sólo resulta útil en caso de que el aventurero haya quedado atrapado en la mazmorra o haya sido engullido por el inodoro.

LAS INSTRUCCIONES

En este momento ya dispones de un juego de aventuras que funciona a la perfección, por lo que ha llegado el momento de darle los últimos toques.

Si no se le dan instrucciones, el aventurero no podrá saber el objetivo de todos tus esfuerzos, ni lo que tiene que hacer. Antes de añadirle a un juego el conjunto de instrucciones, comprueba la cantidad de memoria que te queda disponible. Si queda poca, es hora de eliminar todas las sentencias REM, aunque puede ser que ello te obligue, para evitar errores, a cambiar la numeración de los GOSUB que envían el programa hacia ellas.

La cantidad de instrucciones a incluir es algo que conviene considerar con cuidado. Tienes que tomar una decisión dependiente de la cantidad de memoria disponible, de cuántas sugerencias quieras dar en cada etapa, y de otras consideraciones tales como el formato de la pantalla de tu máquina, que afectará en gran medida al grado de detalle que puedas dar antes de tener que pasar a otra pantalla.

Como la aventura de **INPUT** es muy sencilla, la rutina de instrucciones es corta y contiene poca informacion. Aquí la tienes:

```
10 PRINT"QUIERES VER LAS
   INSTRUCCIONES?"
20 A$=INKEY$:IF A$=""
   THEN 20
30 IF A$="S" THEN GOSUB 6000
6000 REM ** INSTRUCCIONES
6010 PRINT:PRINT"A CAUSA DE
     UNA CRISIS ECONOMICA HAS
     ":PRINT"HUIDO DE TU PAIS
     "
6020 PRINT:PRINT"LA SOLUCION
     A TUS PROBLEMAS ESTA
     EN"
6025 PRINT"ENCONTRAR EL GLOBO
      OCULAR,"
6027 PRINT"PASAR AL FINAL Y
     SUPERAR"
6029 PRINT"LA PRUEBA DE
     INICIATIVA"
6030 PRINT:PRINT"EVITA A TODA
     COSTA AL INSPECTOR
     FISCAL"
6040 PRINT:PRINT"PULSA UNA
     TECLA"
6050 A$=INKEY$:IF A$=""
     THEN 6050
6060 RETURN
```

Ahora ya puedes almacenar (SAVE) en cinta la aventura completa.

En el próximo capítulo, veremos la forma de utilizar la estructura que hemos seguido a lo largo del juego del Ojo Precioso de la Imagen Pùrpura, para que sirva de base a tus propias aventuras.

AVENTURAS: EL SIGUIENTE PASO

Las aventuras se parecen al tabaco: producen hábito; además las puedes comprar ya empaquetadas o *liar* las tuyas propias. Aquí verás la manera de usar la aventura como base para desarrollar la tuya propia.

En este momento tienes almacenado en cinta un juego completo de aventuras que funciona perfectamente. Al ir recorriendo todo su desarrollo, has visto cómo se van combinando todos los elementos que la constituyen, partiendo de un bosquejo muy rudimentario de la historia. En este capítulo veremos la manera de utilizar dicho juego como punto de partida para el desarrollo de tus aventuras domésticas.

Aunque en los anteriores capítulos hemos visto algunas sugerencias sobre posibles variaciones del juego, seguidamente nos ocuparemos con más profundidad de la forma en que lo puedes hacer.

No siempre será posible ser muy específico acerca de las alteraciones a introducir, ya que muchas de ellas dependerán totalmente de la aventura que estés escribiendo, pero muchas de ellas serán muy fáciles de incorporar siguiendo las instrucciones que veremos más adelante. Al principio puede que algunas de las técnicas te desanimen un poco, pero si empiezas tratando de escribir una aventura corta y sencilla, en seguida captarás los principios básicos. Durante las primeras fases no pretendas hacer demasiadas alteraciones al mismo tiempo; es mejor que te limites a ir recorriendo sistemáticamente las secciones de este artículo y no tendrás muchos problemas.

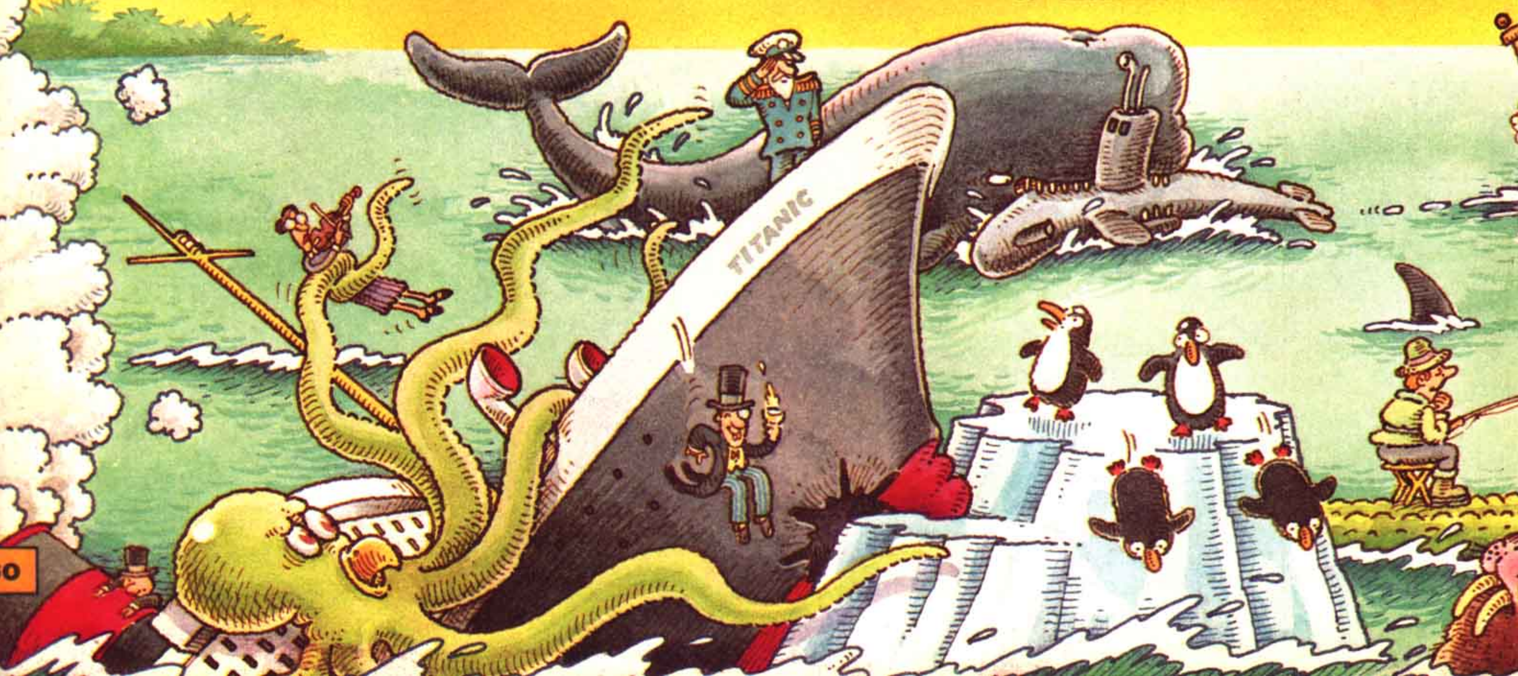
TEMAS PARA TU PROPIA AVENTURA

Antes de ponerte a escribir un juego de aventuras propio, tienes que inventarte un bosquejo de la historia que te sirva como punto válido de partida.

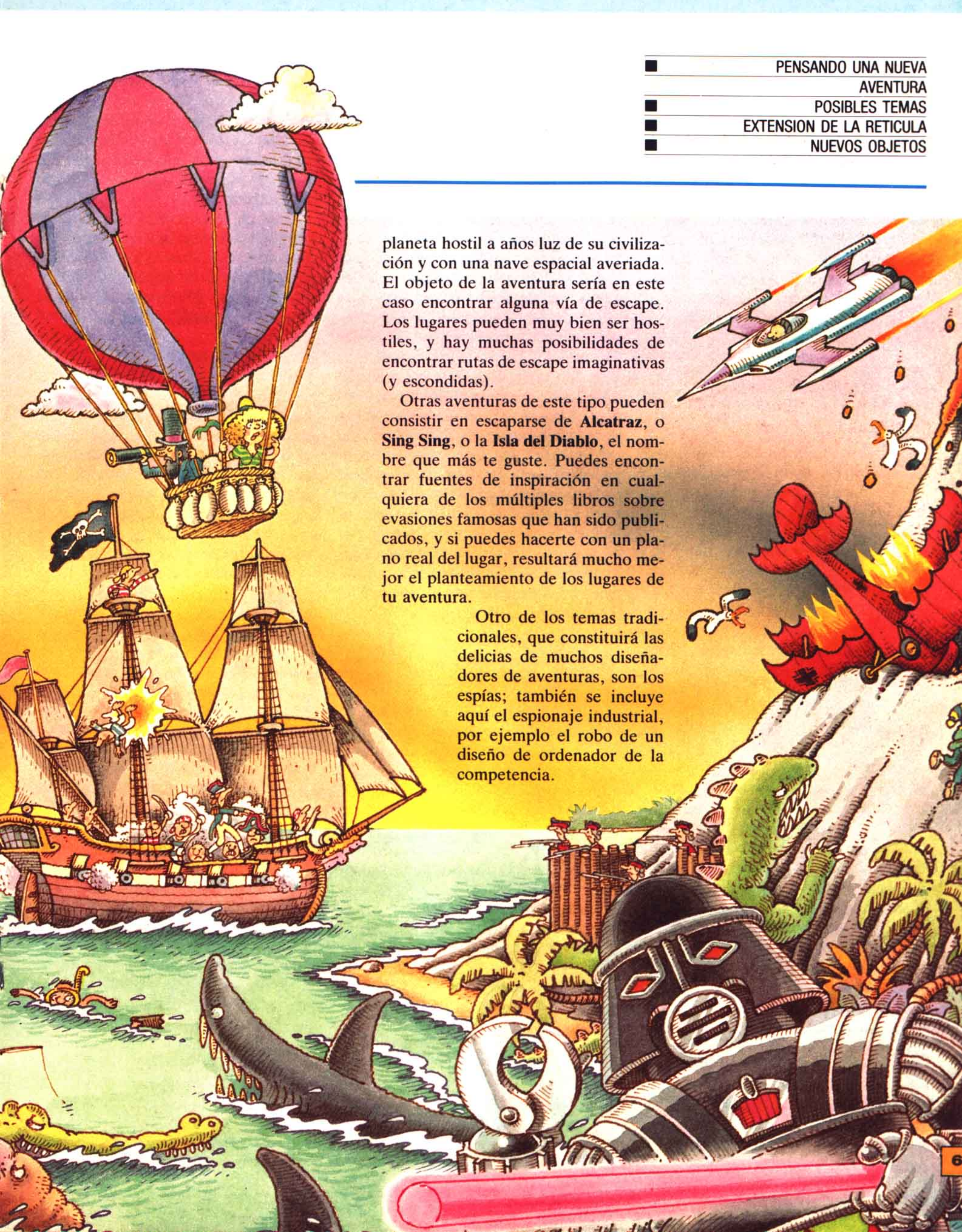
La estructura argumental de las aventuras de mayor éxito suele ser bastante tradicional: hay un principio, una fase intermedia, y un final, con una secuencia impuesta por el orden en que se quiere que aparezcan los enigmas que haya que resolver. Sin embargo es una suerte que no seas una **Agatha Christie** a la hora de escribir juegos de aventuras, ya que aunque hay montañas de ideas posibles, su realización no resulta fácil en las primeras etapas. A continuación presentamos unas cuantas sugerencias que harán más fácil tu labor.

Podrías estructurar la aventura en torno a una novela policíaca. El punto de partida podría ser una habitación en la que hay un cadáver con un puñal clavado, y el objetivo final del juego sería encontrar quién es el asesino. Tal vez prefieras utilizar el personaje de un mayordomo en lugar del inspector de hacienda. Su papel podría consistir en ayudar al aventurero, o en estorbarle.

Hay varias maneras de utilizar un argumento de naufragios en la aventura. Puedes hilvanar una historia tradicional de náufragos y piratas, o hacer que tu aventurero sea el único superviviente de un accidente aéreo. Si prefieres situar tu aventura en el futuro, puedes montar un desastre espacial que haga que el aventurero se encuentre abandonado en un



- PENSANDO UNA NUEVA AVENTURA
- POSIBLES TEMAS
- EXTENSION DE LA RETICULA
- NUEVOS OBJETOS



planeta hostil a años luz de su civilización y con una nave espacial averiada. El objeto de la aventura sería en este caso encontrar alguna vía de escape. Los lugares pueden muy bien ser hostiles, y hay muchas posibilidades de encontrar rutas de escape imaginativas (y escondidas).

Otras aventuras de este tipo pueden consistir en escaparse de **Alcatraz**, o **Sing Sing**, o la **Isla del Diablo**, el nombre que más te guste. Puedes encontrar fuentes de inspiración en cualquiera de los múltiples libros sobre evasiones famosas que han sido publicados, y si puedes hacerte con un plano real del lugar, resultará mucho mejor el planteamiento de los lugares de tu aventura.

Otro de los temas tradicionales, que constituirá las delicias de muchos diseñadores de aventuras, son los espías; también se incluye aquí el espionaje industrial, por ejemplo el robo de un diseño de ordenador de la competencia.

También puedes sacar multitud de temas de la Historia. Por ejemplo las Cruzadas son una evidente fuente de inspiración para los juegos de aventuras; lo mismo puede decirse de cualquier campaña militar.

Por último, un tema que se ha convertido en tópico: una aventura basada en un holocausto nuclear. Las posibilidades son enormes: mutantes, búsqueda de los trajes antirradiación, grupos de bandoleros que merodean muriéndose de hambre, intentando encontrar comida y agua sin contaminar, etc, etc,etc.

¿MAS LUGARES?

La aventura de **INPUT** es mucho más corta de lo que suele ser la longitud normal de estos juegos, por lo que pronto te encontrarás que tus propios juegos de aventuras superan ampliamente a este programa.

Sigue las instrucciones de las páginas 38 a 43 para obtener una retícula adecuada a tu programación. La retícula de **INPUT** tiene 6×4 lugares, en total 24, de los que sólo se utilizan 12. Si decides trabajar adaptándote a esta retícula, puedes hacer una de dos cosas: o bien modificas el programa existente, lo que representa menos trabajo aunque es más difícil de descifrar, o tecleas un programa completamente nuevo, lo cual representa algo más de esfuerzo, aunque puede que te resulte menos confuso. Dependiendo de la elección que hagas, puedes cargar (LOAD) el programa existente desde la cinta, o, si tienes una impresora, listarlo en papel. Las adaptaciones que siguen a continuación dependen del tamaño de la retícula que te resulte útil. Si tienes 24 lugares o menos, puedes usar la retícula existente tal como está, dibujando el mapa sobre la misma. Si tu mapa requiere una retícula mayor, dibújala y numérala de la forma que ya sabes.

Después de organizar la retícula, puedes empezar a introducir tu propio juego de descripciones de lugares en la máquina. Tienen que sustituir a las descripciones de los lugares existentes a partir de la línea 5000.

Cada descripción de un lugar ha de ir seguida con la línea que contiene las posibles salidas del mismo, tal como ocurría con el programa original. Las variables N, S, E y O corresponden a Norte, Sur, Este y Oeste. Pueden tomar los valores 0 y 1; 0 siginifica que no hay salida en esa dirección, mientras que 1 significa que sí hay una salida. El esfuerzo extra de teclear unas cuantas líneas suplementarias de sentencias REM con los números de los lugares es algo que te va a merecer la pena.

El siguiente paso es modificar las sentencias ON ... GOSUB de las líneas 330 a 350. El primer número que sigue a la sentencia GOSUB de la línea 330 es el número de línea en que el ordenador encontrará la descripción del lugar 1. Si no hay un lugar 1 en la aventura —no tienes porqué utilizar todas las casillas de la retícula— se introduce en su lugar un cero. El siguiente número corresponde al número de línea del segundo lugar, y así sucesivamente. Tiene que haber un número para cada uno de los lugares de la retícula.

MOVIMIENTO

Si has diseñado un juego basado en una retícula de tamaño diferente al de la utilizada en la aventura, necesitarás modificar las rutinas de movimiento de las líneas 1000 a 1040. Más específicamente, si la retícula no tiene una anchura de seis cuadros, tendrás que cambiar las líneas de Norte y Sur (líneas 1010 y 1030) , ya que para cambiar de fila lo que hacías era sumar o restar seis. Para hacer la modificación, no tienes más que contar de cuántas casillas se compone la fila de tu retícula, y sustituir el número 6 por el valor del nuevo ancho.

LOS OBJETOS

Los objetos de tu nueva aventura serán diferentes de los de la aventura de **INPUT**, por lo que es probable que tengas que hacer cambios bastante extensos en las líneas 160 a 260.

Cuenta el número de objetos que vayas a utilizar en tu nueva aventura. Este número determina el valor de NB y debe ser el primer dato de la línea 200, siendo utilizado para dimensionar las matrices de la línea 180, y para los bucles FOR ... NEXT de otras partes del programa.

Aunque resulta más claro utilizar una línea de programa separada para cada objeto, si has escrito un juego que utilice muchos objetos, puede que te resulte más cómodo poner más de un objeto en cada línea. Cualquiera que sea la forma en que decidas ponerlos, tus datos deben guardar el orden correcto, ya que cada uno de los tres grupos de datos forman parte de matrices diferentes. El orden es el siguiente: número de lugar, descripción corta y descripción larga. Si el objeto no aparece hasta más tarde en la aventura, tal vez debido a que el aventurero lo encuentra, o es de aparición aleatoria, como es el caso del inspector de hacienda, el correspondiente número de lugar será un cero.

NUEVAS PALABRAS

Haz una lista de todas las instrucciones que el ordenador debe esperar recibir del aventurero durante el juego. Dicha lista incluirá palabras sencillas, tales como las órdenes y las palabras AYUDA e INVENTARIO, y órdenes formadas por dos palabras, tales como COGER LAMPARA o MATAR POSADERO.

Las entradas a base de dos palabras se desdoblan en V$ y N$, verbos y nombres, aunque esta denominación no siempre corresponda estrictamente a la definición gramatical.Tu interés debe centrarse en todas las palabras sencillas y en la primera palabra de cada pareja. Para los fines del programa, las primeras palabras son los verbos, V$. Agrupa los diferentes verbos con arreglo a su significado, por ejemplo COMER y MASTICAR, o bien OLER y HUSMEAR. Cada uno de estos grupos necesitará un número que también deberás anotar. No importa cómo se asigne ese número; basta que sepas que cada número se re-

fiere a un determinado grupo de palabras.

Ya puedes modificar el programa. La rutina de manejo de verbos está en las líneas 110 a 150. Los verbos y sus correspondientes números se introducen como datos en las líneas 140 y 150, como pares, cuyo primer componente es el número y el segundo el verbo.

No te olvides de volver a dimensionar las matrices de la línea 120 y de ajustar el bucle FOR ... NEXT de la línea 130 adaptándolo al número total de objetos que quieras usar.

RUTINAS DE VERBOS

Cada una de las categorías separadas de verbos, es decir, cada uno de los números, requerirá una rutina separada.

Es difícil dar instrucciones explícitas acerca de la forma de escribir estas rutinas, debido a que una buena parte de las rutinas de cualquier aventura no valdrá para las demás.

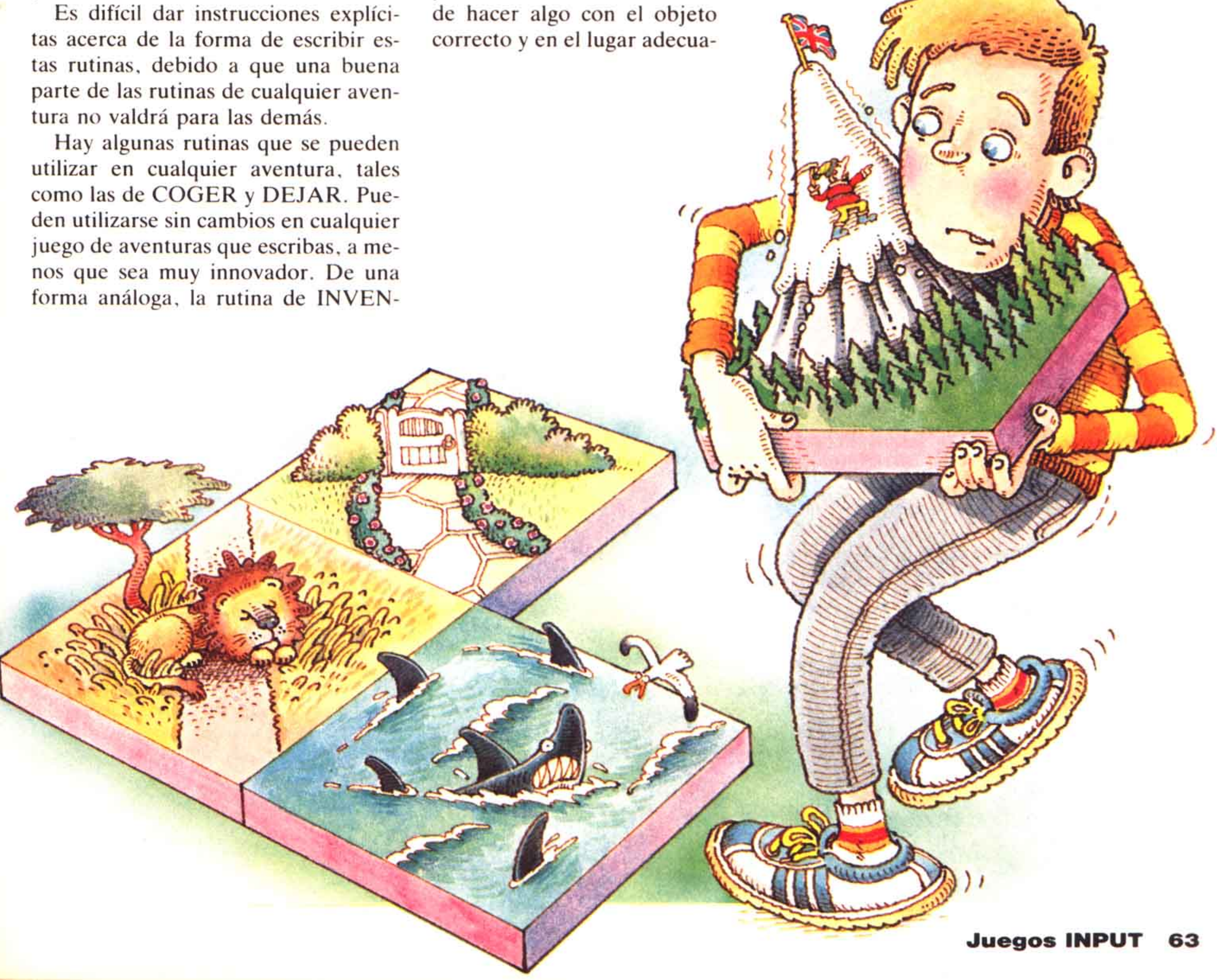
Hay algunas rutinas que se pueden utilizar en cualquier aventura, tales como las de COGER y DEJAR. Pueden utilizarse sin cambios en cualquier juego de aventuras que escribas, a menos que sea muy innovador. De una forma análoga, la rutina de INVENTARIO (líneas 1070 a 1130) es la misma para cualquier aventura, por lo que puedes utilizarla sin cambios en la medida en que la matriz es la misma y que NB —el número de objetos— tiene el mismo significado en la nueva aventura.

Otra rutina, que podría ser de aplicación sería la rutina de encendido de la lámpara, porque el encender y apagar lámparas y linternas es una ocupación muy frecuente en los juegos de aventuras. Dicha rutina está situada en las líneas 1490 a 1530.

Probablemente las demás rutinas no son lo bastante generales como para trasladarlas en bloque, pero hay algunos puntos que conviene que tengas en cuenta cuando escribas tus propias rutinas de verbos. Básicamente las rutinas se ponen ahí para comprobar si el aventurero trata de hacer algo con el objeto correcto y en el lugar adecuado. Si el lugar está equivocado, el programa presentará un mensaje de que eso es apropiado para ciertas situaciones, pero AHI NO. Ocurra lo que ocurra, cerciórate de que el aventurero conoce cuál fue el efecto de su última instrucción, en otras palabras, para cualquier cosa que se le diga a la máquina que haga, debe aparecer en pantalla un mensaje de respuesta.

Cuando tengas lista tu rutina de verbos, introdúcela en el programa. Si numeras el programa de forma análoga a la aventura de **INPUT**, el lugar para esta rutina estará entre las líneas 1070 y 2999.

El ordenador tiene que poder selec-



cionar la rutina correcta de acuerdo con el verbo utilizado por el aventurero. Para que pueda hacer esto tienes que modificar la línea 510.

Lo único que tienes que hacer para ello es observar tu lista de números de verbos. A continuación, utilizando ese orden numérico, pon después de la sentencia ON ... GOTO, las líneas de comienzo de la rutina correspondiente a cada verbo.

RUTINA DE AYUDA

La rutina final a la que debes dedicar tu atención es la de AYUDA. Considera en qué puntos de tu aventura podría ser necesaria una sugerencia, y utiliza una línea IF ... THEN para hacerla.

Hay otros detalles que puede que requieran modificación, dependiendo de las características de tu aventura; tal es el caso de la línea 320, que hace que aparezca el inspector de hacienda. No pierdas de vista tampoco el lugar de comienzo, que se establece en la línea 280.

VARIABLES Y MATRICES

Ahora que ya sabes «meterte dentro» del programa de la aventura, aquí tienes una lista de las variables y matrices junto con el uso a que se destinan:

R$() matriz de verbos y respuestas.

R() matriz de números de respuestas.

Los elementos correspondientes de las dos matrices anteriores son los pares de verbos y los significados.

OB() matriz con el número de lugar para cada objeto.

OB$() matriz de descripciones cortas de los objetos.

SI$() matriz de descripciones largas de los objetos.

Los elementos correspondientes de las tres matrices anteriores contienen información relativa a cada objeto en particular.

NB número de objetos de la aventura. Se utiliza para dimensionar las matrices y en los bucles FOR ... NEXT.

L situación actual del aventurero.

LA indicador de estado de la lámpara. Se pone a 1 cuando está encendida y a 0 cuando está apagada.

TA indicador del inspector de hacienda.

N,S,E,O direcciones de salida. Se ponen a 1 si existe una salida en esa dirección y a 0 si no existe.

I$ entrada total antes de ser desglosada en verbos y nombres.

V$ parte de verbos de I$.

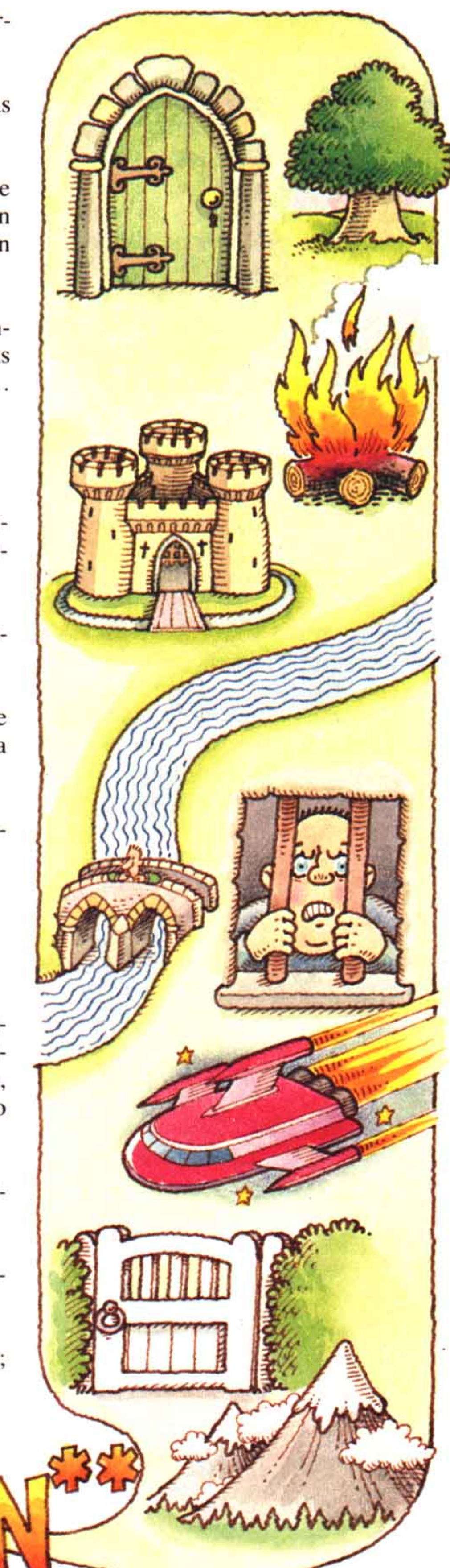
N$ parte de nombres de I$.

I número correspondiente al significado de un determinado verbo. Se utiliza para dirigirse a la rutina correcta, que es la que se ocupa de ese verbo en particular.

IN número de objetos del INVENTARIO.

A$ respuesta a la pregunta QUIERES PROBAR OTRA VEZ?

G número de objeto abandonado; G es un elemento de la matriz OB.



REM**LOCATION**

LA MEMORIA DE VIDEO DE MSX. LOS REGISTROS DEL V.D.P.

El control de la memoria de vídeo y de la imagen de pantalla recae sobre el *chip* VDP. Vamos a concluir esta serie de artículos sobre la memoria de vídeo con un análisis en profundidad de los registros de este *chip* (*VIDEO DISPLAY PROCESSOR*).

En realidad los registros del VDP tienen escaso interés para la programación en BASIC. Sin embargo pueden serte útiles si programas en código máquina, ya que son el único medio de, por ejemplo, cambiar el modo de pantalla mediante el uso de las rutinas de la BIOS (*Basic Input Output System*).

El VDP (*Video Display Processor*) es un procesador (lo mismo que el Z80) que llevan todos los ordenadores **MSX** para el manejo de la memoria de vídeo. Esto permite al procesador Z80 no tener que ocuparse de esa función y tener más tiempo para otras cosas.

El nombre completo del VDP es TMS9918 y ha sido fabricado por **Texas Instruments** (El VDP de los ordenadores de la segunda generación **MSX** es diferente a éste). Al programar en C.M. necesitamos acceder al VDP tanto para escribir en sus registros como para leer o escribir en los bytes de la VRAM.

El VDP tiene 9 registros de los que 8 son de sólo escritura, es decir, en ellos sólo podemos escribir, y uno de sólo lectura, en el que sólo podemos leer. No obstante, el sistema operativo guarda una copia de los 8 registros de escritura en el área de trabajo del sistema, por lo que podemos «leer» lo que hay en esos 8 registros. Al programar en C.M. es conveniente que anotes los valores que toman los registros en cada momento, porque no es posible leerlos. (Si no entiendes algo de lo que te estamos diciendo, no te preocupes; hablaremos del área de trabajo del sistema al hablar de la BIOS).

Veremos ahora para qué sirve cada uno de dichos registros.

EL REGISTRO 0

Los 8 registros de sólo escritura van numerados del 0 al 7. Podemos acceder a los registros del VDP mediante la variable reservada del BASIC VDP (1) en la que I puede valer 0, 1, ..., 8. Podemos escribir en los registros 0 al 7 haciendo

VDP(I) = Expresión

donde Expresión es cualquier expresión válida en BASIC. También podemos leer lo que hay en los 9 registros mediante

PRINT VDP(I)

(En realidad, como ya hemos dicho, no leemos los registros, sino la copia de los mismos que el sistema mantiene en el área de trabajo).

Cada registro del VDP tiene 8 bits. El programa que adjuntamos te permitirá conocer en cualquier momento el contenido de los 9 registros del VDP. El registro 0 puede tomar los valores de la figura. En esta figura, todos los ceros indican que tales bits deben estar SIEMPRE a 0. El bit 0 está preparado para la conexión de un VDP externo, de otro ordenador. Puedes hacer este bit igual a 1, pero lo único que conseguirás es que se distorsione la imagen en el monitor. Como ves, en principio poco interés tiene este bit, sin embargo, te permite utilizar el VDP de otro ordenador, y por tanto su monitor.

El bit 1 (M3) tomará el valor 1 cuando estés trabajando en SCREEN 2. Si pones este bit a 1 estando en SCREEN 0, sólo conseguirás que te aparezcan cosas extrañas en la pantalla. No basta con hacer este bit igual a 1 para estar en SCREEN 2. Hace falta, además actualizar todos los registros del VDP, y los 16K de la VRAM. Por tanto este bit sólo tiene interés al programar en C.M.

EL REGISTRO 1

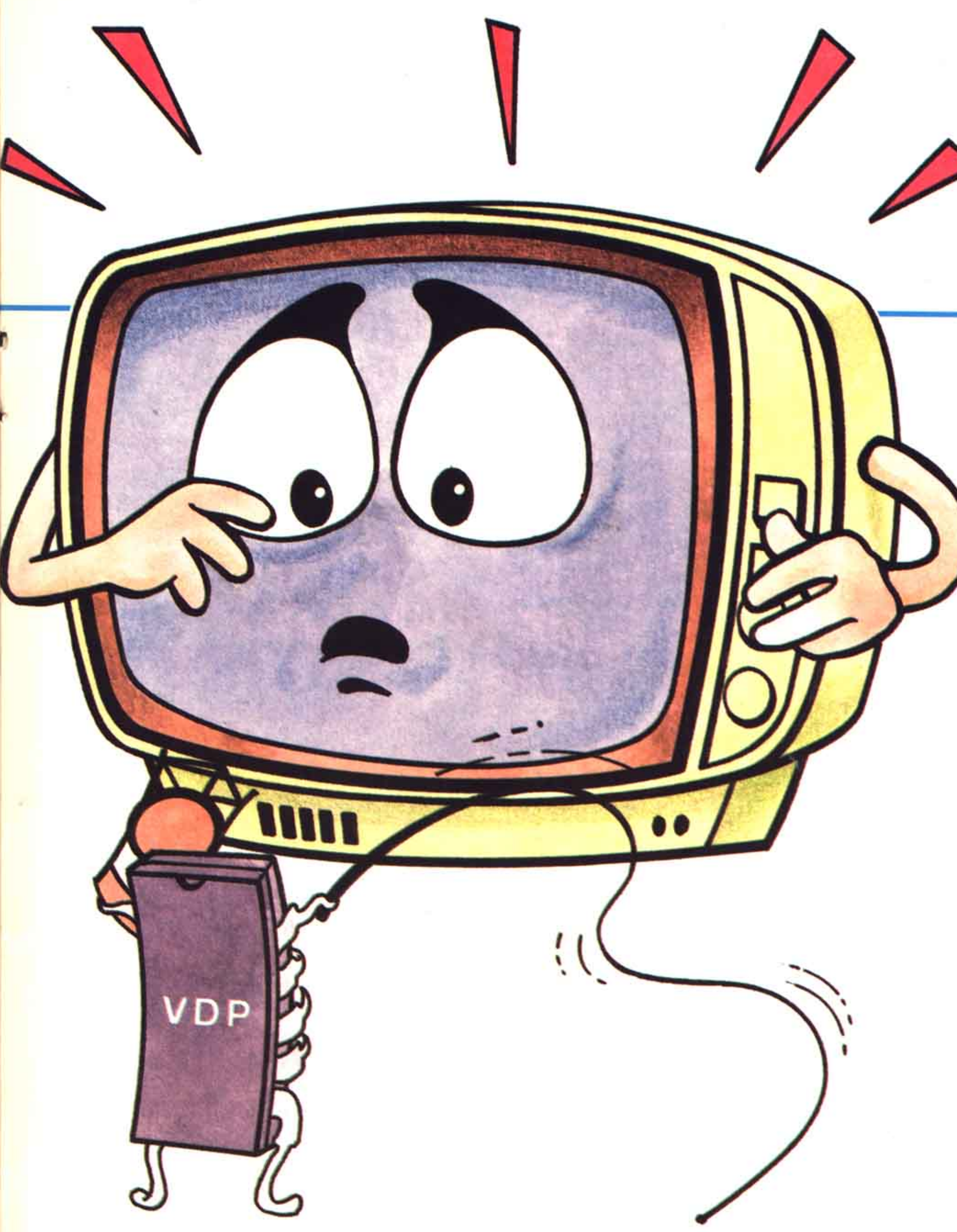
De los 8 bits del registro 1, sólo el bit nº 2 debe mantenerse a 0 (ver figura). Las funciones que realizan el resto de los bits son las siguientes:

Bit 0 (MAG): Define el tamaño de *sprites*. Si MAG=0, los puntos del *sprite* tienen un tamaño 1 × 1 (sin ampliar), y si MAG=1, tienen un tamaño 2 × 2 (ampliados). Prueba el programa 2 y verás para qué sirve el bit MAG. Esto no lo podías hacer con la instrucción SCREEN, ya que al ejecutarse ésta, se borra de la memoria de vídeo toda la información referente a *sprites*.

PROGRAMA 2

```
10 SCREEN 1
20 FOR I=1 TO 8
30 READ A
40 A$=A$+CHR$(A)
50 NEXT
60 SPRITE$(1)=A$
70 PUT SPRITE 1,(100,100),
   15,1
80 VDP(1)=VDP(1) OR 1
90 FOR I=1 TO 200
100 NEXT
110 VDP(1)=VDP(1) XOR 1
120 FOR I=1 TO 200
130 NEXT
140 GOTO 80
150 DATA 255,129,129,129,
    129,129,129,255
```

Bit 1 (SIZE): Determina si el *sprite* es de 8 × 8 ó de 16 × 16. Si SIZE = 0, todos los *sprites* serán de 8 × 8 y si SIZE = 1 serán de 16 × 16. Como en el caso anterior, también puedes cambiar el valor de SIZE dentro de un



programa sin necesidad de utilizar SCREEN. Ejecuta el programa 3.

PROGRAMA 3

```
10 SCREEN 1
20 FOR I=0 TO 3
30 FOR J=1 TO 8
40 READ A
50 A$(I)=A$(I)+CHR$(A)
60 NEXT
70 SPRITE$(I)=A$(I)
80 NEXT
90 PUT SPRITE 0,(100,100)
   ,15,0
100 VDP(1)=VDP(1) OR 2
110 FOR I=1 TO 200
120 NEXT
130 VDP(1)=VDP(1) XOR 2
140 FOR I=1 TO 200
150 NEXT
160 GOTO 100
170 DATA 255,128,128,128,128,
    128,128,128,128,128,128,
    128,128,128,128,255,255,
    1,1,1,1,1,1,1,1,1,1,1,1,
    1,1,255
```

Tampoco podías hacer esto con la instrucción SCREEN por la misma razón que antes.

Bit 3 (M2): Este bit será igual a 1 si estamos en SCREEN 3. En caso contrario será igual a 0.

Bit 4 (M1): Será 1 en SCREEN 0 y 0 en caso contrario. Los tres bits M1, M2 y M3 (este último del registro 0) informan al VDP de cuál es el modo de pantalla actual. En resumen:

	M1	M2	M3
SCREEN 0	1	0	0
SCREEN 1	0	0	0
SCREEN 2	0	0	1
SCREEN 3	0	1	0

Bit 5 (IE): Este bit es muy interesante porque desactiva las interrupciones. Es decir, si IE=0, las interrupciones están desactivadas. De esta forma, el ordenador no leerá el teclado y no habrá forma de interrumpir un programa. Sólo conseguirás que el ordenador te responda haciendo un RESET o haciendo que IE sea igual a 1 en alguna parte de tu programa.

Para desactivar las interrupciones puedes hacer:

VDP(1) = VDP(1) XOR 32

y para volverlas a activar:

VDP(1) = VDP(1) OR 32

Bit 6 (BL): Cuando, estando en SCREEN 2, dibujas, por ejemplo, un círculo y lo rellenas de un color, puedes observar cómo se ejecutan las instrucciones porque el dibujo se realiza con cierta lentitud. Sin embargo, puedes hacer que el dibujo aparezca instantáneamente haciendo uso del bit BL. Cuando BL=0, la pantalla no está activa y, aunque todo lo que envíes a la pantalla se escribirá en la VRAM, tú no lo verás hasta que vuelvas a activarla haciendo BL=1. Para desactivar la pantalla usa la instrucción:

VDP(1) = VDP(1) XOR 64

y para volverla a activar:

VDP(1) = VDP(1) OR 64

PROGRAMA 4

```
10 SCREEN 2
15 VDP(1)=VDP(1) XOR 96
20 CIRCLE (100,100),
   70,,,,1.4
30 PAINT (100,100)
40 VDP(1)=VDP(1) OR 96
50 GOTO 50
```

El programa 4 hace lo que te acabamos de explicar. Observarás que hemos hecho:

VDP(1) = VDP(1) XOR 96

para desactivar la pantalla. En realidad lo que hemos hecho es desactivar la pantalla y las interrupciones (32+64=96), ya que si no desactivas las interrupciones, puedes pulsar CTRL-STOP con la pantalla desactivada y no podrás volver a activarla. (Si te ocurre esto escribe, aunque tu no lo veas: VDP(1) = VDP(1) OR 64, con lo que volvería a activarse la pantalla).

Por último, el **bit 7 (VRAM)** informa al VDP del tamaño de la memoria VRAM instalada. Si VRAM=0, la VRAM es de 4 Kbytes y si VRAM=1, la VRAM es de 16K. No te conviene modificar este bit.

EL REGISTRO 2

Este registro contiene menos información que el registro 1. El número que contiene el registro 2, multiplicado por 1024 (2 ↑ 10), da como resultado la dirección de comienzo de la tabla de nombres. Dicho número es siempre menor que 16, por lo que sólo se hace uso del *nibble* bajo del byte correspondiente al registro 2. Los valores que toma este registro inicialmente son:

SCREEN 0 0
SCREEN 1 6
SCREEN 2 6
SCREEN 3 2

VDP (2) te permite, por ejemplo hacer uso de la memoria VRAM que no se utiliza en SCREEN 0. Por ejemplo, después de ejecutar el programa 5, escribe (en SCREEN 0)

VDP(2) = 14

PROGRAMA 5

```
10 FOR I=0 TO 23
20 READ A
30 VPOKE 14336+I,A
40 NEXT
50 DATA 69,110,116,130,114,
   97,116,101,32,99,111,110,
   32,34,73,110,112,117,116,
   32,77,83,88,34
```

Verás que en la parte superior izquierda de la pantalla aparece nuestro mensaje. Teclea ahora (aunque no lo veas) VDP(2)=0, y habrás vuelto a la pantalla normal.

EL REGISTRO 3

Si multiplicas el número almacenado en este registro por 64 (2 ↑ 6) tendrás la dirección de comienzo de la tabla de color. Esto sólo es práctico en SCREEN 1. En este modo de pantalla es, normalmente

VDP(3) = 128

EL REGISTRO 4

Este registro tiene información sobre la tabla del generador de patrones del modo de textos o de gráficos. Si multiplicas VDP(4) por 2048 (2 ↑ 11) tendrás dicha dirección, que en cada uno de los modos es:

SCREEN	VDP(4)	VDP(4)*2048
0	1	2048
1	0	0
2	3	6144
3	0	0

Observa el efecto que produce el programa 6. En las líneas 20 a 40 del mismo copiamos la tabla del generador de patrones, que comienza en la dirección 2048 de la VRAM, en el bloque de memoria VRAM que comienza en la dirección 4096, pero «invirtiéndolos» (línea 30). Es decir, hemos creado una nueva tabla del generador de patrones en la que todos ellos son los inversos de los que figuran en la tabla original. Las sentencias comprendidas entre las líneas 80 y 140 te permiten ver el efecto en la pantalla.

PROGRAMA 6

```
10 SCREEN 0
20 FOR I=2048 TO 4095
30 VPOKE 2048+I,VPEEK(I)
   XOR 255
40 NEXT
50 FOR I=0 TO 959
60 VPOKE I,65
70 NEXT
80 VDP(4)=2
90 FOR I=1 TO 200
100 NEXT
110 VDP(4)=1
120 FOR I=1 TO 200
130 NEXT
140 GOTO 80
```

EL REGISTRO 5

Contiene información sobre la dirección de comienzo de la tabla de atributos de *sprites*, de la que hablamos en el número 5 de **INPUT**. Debes multiplicar su valor por 128 (2 ↑ 7). Así tenemos para los modos 1, 2 y 3: VDP(5) = 54, es decir que la dirección de comienzo de la tabla de atributos de los *sprites* es 54*128=6912.

PROGRAMA 7

```
10 SCREEN 1
20 FOR I=1 TO 8
30 READ A
40 A$=A$+CHR$(A)
50 NEXT
60 FOR I=0 TO 3
70 SPRITE$(I)=A$
80 NEXT
90 FOR I=0 TO 3
100 PUT SPRITEI,(10+I*10,10)
    ,15,I
110 NEXT
120 FOR I=0 TO 15
130 READ A
140 VPOKE 4096+I,A
150 NEXT
160 VDP(5)=54
170 VDP(5)=32
180 GOTO 160
1000 DATA 255,129,129,129,129
     ,129,129,255
2000 DATA 10,50,0,15,10,60,1,
     15,10,70,2,15,10,80,3,15
```

Teclea el programa 7. Observarás que lo que hacemos es crear una nueva tabla de atributos de *sprites* que comienza en la dirección 4096 de VRAM

y, con los DATA de la figura 2000 hacemos que los 4 *sprites* definidos anteriormente se situen en 4 posiciones a la derecha de las posiciones iniciales (líneas 90 a 110). En las líneas 160 y 170 activamos la tabla original (VDP(5)=54) y la nueva tabla (VDP(5)=32), con lo que en la pantalla vemos ¡¡¡8 SPRITES EN LA MISMA LINEA!!!

Lamentablemente se observa un cierto centelleo en la presentación en pantalla.

EL REGISTRO 6

El valor de VDP(6) multiplicado por 2048 (2 ↑ 11) nos dará la dirección de comienzo de la tabla del generador de modelos de *sprites*, que tanto en el modo 1 como en el modo 2 y en el 3 es 14336. Por tanto VDP(6)=7 en estos tres modos.

EL REGISTRO 7

Si dividimos este registro en sus dos *nibbles*, el alto contiene el color de la tinta en el modo de texto SCREEN 0, que inicialmente es 15=blanco, y el bajo contiene el color del fondo en todos los modos.

EL REGISTRO 8

Se denomina también «registro de estado». Es el registro de sólo lectura que mencionábamos al principio.

Los bits 0 a 4 (ver figura) de este registro contienen el número del quinto *sprite* cuando se han situado 5 *sprites* en una misma línea horizontal. El bit 5 vale 1 si ha habido colisión entre *sprites* y 0 en caso contrario. El bit 5 está a 1 si hay 5 *sprites* en una línea horizontal. En caso contrario está a 0. Y por último, el bit 7 sirve para controlar las interrupciones y vale 1 cuando el VDP ha terminado de analizar la pantalla y 0 después que el VDP ha leído el registro de estado.

Para terminar con los registros del VDP, veamos cómo acceder a ellos en código máquina.

7	6	5	4	3	2	1	0	
0	0	0	0	0	0	M3	VE	REGISTRO 0
VRIM	BL	IE	M1	M2	0	SIZE	MAG	REGISTRO 1
0	0	0	0	0/1 TABLA DE NOMBRES	0/1	0/1	0/1	REGISTRO 2
0/1 TABLA DE COLOR	0/1	0/1	0/1	0/1	0/1	0/1	0/1	REGISTRO 3
0	0	0	0	0	0/1 T. GENER.R PATRONES	0/1	0/1	REGISTRO 4
0	0/1 TABLA DE ATRIBUTOS DE SPRITES	0/1	0/1	0/1	0/1	0/1	0/1	REGISTRO 5
0	0	0	0	0	0/1 T. GENERADOR SPRITES	0/1	0/1	REGISTRO 6
0/1 COLOR TINTA	0/1	0/1	0/1	0/1 COLOR FONDO	0/1	0/1	0/1	REGISTRO 7
INTERR.S	5 SPRITES	COLISION SPRITES	0/1 NUMERO DEL 5.° SPRITE	0/1	0/1	0/1	0/1	REGISTRO 8

LOS REGISTROS DEL VDP

Para escribir un valor en un registro del VDP debes hacer lo siguiente:

```
IN    A,(99H)
LD    A,N
OUT   (99H),A
LD    A,r
OUT   (99H),A
```

donde n es el valor que queremos escribir en el registro r y r el número del registro (de 0 a 7) + 128 (r=0..7 + 128).

No puedes leer los registros del VDP desde código máquina, salvo el registro de estado, lo que se consigue con

IN A,(99H)

que coloca en el acumulador una copia de dicho registro.

PROGRAMA 8

```
5 CLEAR 200,55000!
10 SCREEN 1
20 FOR I=1 TO 8
30 READ A
40 A$=A$+CHR$(A)
50 NEXT
60 FOR I=0 TO 3
70 SPRITE$(I)=A$
80 NEXT
90 FOR I=0 TO 3
100 PUT SPRITEI,(10+I*10,10)
    ,15,I
110 NEXT
120 FOR I=0 TO 15
130 READ A
140 VPOKE 4096+I,A
150 NEXT
160 FOR I=1 TO 30
170 READ A$
180 POKE 55000!+I,VAL
    ("&H"+A$)
190 NEXT
200 DEFUSR=55001!
210 U=USR(0)
1000 DATA 255,129,129,129,129
     ,129,129,255
2000 DATA 10,50,0,15,10,60,1,
     15,10,70,2,15,10,80,3,15
3000 DATA F3,DB,99,3E,20,D3,9
     9,3E,85,D3,99,3E,04,3D,2
     0,FD,3E,36,D3,99,3E,85,D
     3,99,CD,B7,00,30,E6,C9
```

```
PROGRAMA 9
                                  ORG   55001
                                  LOAD  55001
D6D9 F3                           DI
D6DA DB99                         IN    A,(99H)
D6DC 3E20       COMIENZO:         LD    A,32            ;VDP(5)=32
D6DE D399                         OUT   (99H),A
D6E0 3E85                         LD    A,133
D6E2 D399                         OUT   (99H),A
D6E4 3E04                         LD    A,4             ;BUCLE RETARDO
D6E6 3D         BUCLE:            DEC   A
D6E7 20FD                         JR    NZ,BUCLE
D6E9 3E36                         LD    A,54            ;VDP(5)=54
D6EB D399                         OUT   (99H),A
D6ED 3E85                         LD    A,133
D6EF D399                         OUT   (99H),A
D6F1 CDB700                       CALL  0B7H            ;([CTRL-STOP] PULSADA?
D6F4 30E6                         JR    NC,COMIENZO     ;NO -> COMIENZO
D6F6 C9                           RET                   ;SI -> FIN
                                  END
```

El programa 8 utiliza la rutina en C.M. listada en el programa 9 y hace lo mismo que el programa 7, es decir, situa 8 *sprites* en una misma línea. La rutina hace que prácticamente se vean los 8 *sprites* sin interrupción (sigue habiendo un ligero centelleo). El bucle de retardo de la rutina es para que permanezcan el mismo tiempo en la pantalla las dos tablas de atributos. La rutina sólo puede ser interrumpida al pulsar simultáneamente las teclas CTRL-STOP.

```
10 'MANIPULADOR VDP
20 '===============
30 '
40 'Versi"n 01.061086 - 1537
   bytes
50 '
60 SC=0
70 WIDTH 39
80 KEY1,"screen"
90 KEY2,"vdp"
100 KEY3,"restaur"
110 KEY4,"fin"
120 ON KEY GOSUB 580,660,810,
    930
130 KEY(1) ON
140 KEY(2) ON
150 KEY(3) ON
160 KEY(4) ON
170 ON STOP GOSUB 930
180 STOP ON
190 GOSUB 520
200 X=0
210 Y=0
220 SCREEN 0
230 'MENU
240 '----
250 CLS
260 FOR I=0 TO 8
270 PRINT USING "§        §
    ###   §§   VDP #";V$(I);
    V(I);RIGHT$("0"+HEX$
    (V(I)),2);I
280 NEXT I
290 LOCATE 0,10
300 PRINT "Screen";SC
310 LOCATE 0,0,1
320 ON STRIG GOSUB 470
330 STRIG(0) ON
340 S=STICK(0)
350 FOR I=1 TO 50:NEXT
```




```
360 IF S=0 THEN 340
370 IF S=1 THEN Y=Y-1
380 IF S=3 THEN X=X+1
390 IF S=5 THEN Y=Y+1
400 IF S=7 THEN X=X-1
410 IF Y<0 THEN Y=0
420 IF Y>7 THEN Y=7
430 IF X<0 THEN X=0
440 IF X>7 THEN X=7
450 LOCATE X,Y
460 GOTO 340
470 IF MID$(V$(Y),X+1,1)="0"
    THEN V(Y)=V(Y) OR 2^(7-X)
    ELSE V(Y)=V(Y) XOR
    2^(7-X)
480 V$(Y)=RIGHT$("00000000"+
    BIN$(V(Y)),8)
490 LOCATE 0,Y
500 PRINT USING "§       §
    ###   §§    VDP #";V$(Y);
    V(Y);RIGHT$("0"+HEX$
    (V(Y)),2);Y
510 RETURN 450
520 LOCATE 0,0,0
530 FOR I=0 TO 8
540 V(I)=VDP(I)
550 V$(I)=RIGHT$("00000000"+
    BIN$(V(I)),8)
560 NEXT I
570 RETURN
580 'Key 1
590 CLS
600 INPUT"Introduce nuevo
    modo de pantalla";SC
610 SCREEN SC
620 GOSUB 520
630 SCREEN 0
640 RETURN 200
650 'key 2
660 DEFUSR=&H156
670 U=USR(0)
680 LOCATE ,,0
690 CLS
700 LOCATE 6,12
710 PRINT "(((ESTAS SEGURO
    ??? ---> S/N"
720 W$=INKEY$:IF W$=""
    THEN 720
730 IF W$="S" OR W$="s"
    THEN 750
740 RETURN 250
750 SCREEN SC
760 FOR I=0 TO 7
770 VDP(I)=VAL("&b"+V$(I))
780 NEXT I
790 'IF INKEY$="" THEN 12085
800 RETURN 200
810 'key 3
820 VDP(0)=0
830 VDP(1)=240
840 VDP(2)=0
850 VDP(3)=0
860 VDP(4)=1
870 VDP(5)=0
880 VDP(6)=0
890 VDP(7)=244
900 KEY(3) ON
910 SC=0
920 RETURN 190
930 'key 4
940 DEFUSR=&H3E
950 U=USR(0)
960 LOCATE ,,0
970 SCREEN 0
980 END
```

DESENREDA TUS CADENAS

- COMPARACION Y ORDENACION DE CADENAS
- TROCEADO
- USO DE CADENAS EN EL PROCESO DE TEXTOS

Las cadenas de caracteres se utilizan en toda clase de programas; de hecho se usan en todas las aplicaciones que requieran algo más que simples números. Aquí presentamos unas cuantas maneras de sacar de ellas el máximo provecho.

Una cadena está constituída por una colección de caracteres. Puede tratarse de letras, números, signos de puntuación o cualquier otro de los símbolos del teclado. Además puedes ponerlas juntas en el orden que quieras.

Naturalmente lo normal es que una cadena contenga algún tipo de información útil. Por ejemplo, la cadena «PEDRO DIAZ 241061 S» contiene el nombre, el primer apellido,la fecha de nacimiento y el estado civil de una determinada persona. En total cuatro elementos de información. A su vez la fecha de nacimiento se puede subdividir en día, mes y año, por lo que en realidad hay un total de seis elementos de información.



Son muchas las ocasiones en las que suele ser necesario subdividir (trocear) una cadena de caracteres para extraer determinadas porciones de información, tal como ocurre con la fecha del ejemplo anterior. En otras ocasiones puede ser necesario sumar las cadenas. Puede que a veces también quieras medir la longitud de la cadena o calcular el valor de alguna de sus partes numéricas. Todo esto se puede hacer utilizando unas cuantas **palabras clave** del BASIC.

La suma de cadenas, la llamada concatenación, es la más fácil de estas operaciones. Para ejecutarla no tienes más que utilizar el símbolo +. Si por ejemplo A$ es igual a «HE » y B$ es igual a «AQUI», entonces A$ + B$ es igual a «HE AQUI». La concatenación es una operación que se limita a juntar unas cadenas de caracteres con otras; debe quedar claro que pone unas a continuación de otras, pero no suma sus valores numéricos. Así, «439» + «241» es igual a «439241» y no a 680.

COMPARACION DE CADENAS

El ordenador, además de poner unas cadenas junto a otras, es capaz de compararlas para ver si son iguales, como ocurre en este juego de adivinanzas:

```
10 G=1:R=RND(-TIME):ON
   INT(RND(1)*6)+1
   GOTO 20,30,40,50,60,70
20 B$="MANZANA":GOTO 80
30 B$="NARANJA":GOTO 80
```

```
40 B$="PLATANO":GOTO 80
50 B$="LIMON":GOTO 80
60 B$="MELON":GOTO 80
70 B$="PERA":GOTO 80
80 PRINT CHR$(12);"SOY UNA
   FRUTA, CUAL SOY";
90 INPUT A$
100 IF A$=B$ THEN 155
110 G=G+1
120 PRINT "MAL!"
130 FOR J=1 TO 2000
140 NEXT J
150 GOTO 80
155 IF G=1 THEN PRINT
    "ACERTASTE A LA PRIMERA":
    END
160 PRINT "ACERTASTE TRAS";G;
    " INTENTOS"
```

La línea 10 pone a 1 el contador de adivinanzas y a continuación lanza un dado electrónico para elegir una fruta. Las diferentes opciones están almacenadas en las líneas 20 a 70. Cualquiera que sea la línea a la que vaya el ordenador, el nombre de la fruta correspondiente se almacena en B$, pasando después a la línea 80. Ahora tienes que acertar cuál es la fruta elegida e introducir tu conjetura mediante una sentencia INPUT con la variable A$. Esta variable se compara con B$ y si hay coincidencia total entre ambas, el ordenador presenta el mensaje «HAS ACERTADO UNA VEZ» o bien «HAS ACERTADO» tantas «VECES» como corresponda en cada caso. Si A$ no es igual a B$, el ordenador escribe «ERROR» y tienes que ensayar de nuevo una respuesta. El programa sigue adelante hasta que la respuesta sea la correcta.

Fíjate bien en que aunque la respuesta elegida por tí sea la correcta, la condición A$ = B$ no se cumplirá a menos que la ortografía sea también correcta. Para que sea correcta, las dos cadenas de caracteres han de ser absolutamente idénticas, tanto las letras, como los números y los signos de puntuación.

Puedes utilizar esta técnica de comparación de cadenas de caracteres para comprobar las entradas introducidas desde el teclado, con una lìnea tal como la siguiente:

IF A$ = «SI» THEN PRINT «ESTAS SEGURO?»

Observa que la condición anterior no se satisface si la palabra «sí» se teclea con letras minúsculas.

ORDENACION DE CADENAS

También puedes comparar las cadenas de caracteres utilizando los signos de desigualdad < y >. Podrías usarlos en una línea como la siguiente:

IF A$<B$ THEN PRINT «LA PRIMERA ES »;A$

En este caso en la condición A$ < B$ se examina si la cadena A$ se sitúa antes o después que B$ cuando ambas se colocan en orden alfabético. Pero, ¡cuidado!, el ordenador hace la ordenación alfabética atendiendo al código ASCII de cada letra: a la A le corresponde el código ASCII 65 y a la Z el 90. El problema es que las letras minúsculas también tienen sus códigos ASCII: a la a le corresponde el 97 y a la z el 122. De esta forma, todas las cadenas de caracteres que empiezan por letras mayúsculas se sitúan por delante de las otras.

Y lo que es peor, los números, los signos de puntuación, los espacios y demás signos, también tienen sus códigos ASCII, por lo que el orden resultante para las cadenas puede quedar totalmente trastocado. No obstante, utilizando cuidadosamente los signos < y > se puede obtener una ordenación alfabética de las cadenas de caracteres.

TROCEADO DE CADENAS

Se puede extraer un carácter o un conjunto de caracteres del interior de una cadena. Esto se hace utilizando las funciones LEFT$, RIGHT$ y MID$.

LEFT$(A$,numero) empieza por el principio o extremo izquierdo de la cadena A$ y te da el número de caracteres que especifiques. Si la cadena A$ es «SR JUAN PEREZ» y especificas dos caracteres, es decir, si tecleas LEFT$(A$,2), el resultado será «SR».

Análogamente la función RIGHT$ empieza a contar por el otro extremo de la cadena. Así, RIGHT$(A$,5) dará como resultado «PEREZ».

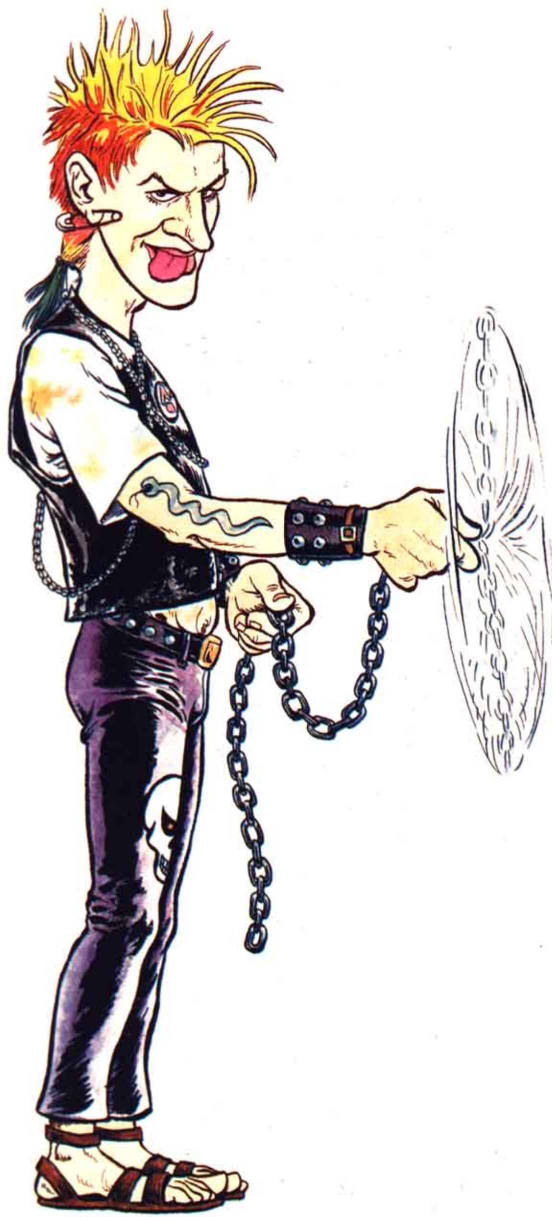
Con la función MID$ puedes especificar dos números, que corresponden a la posición de partida y al número de caracteres que ha de contener el trozo. Por ejemplo, MID$(A$,4,6) empezará en el cuarto carácter J y tomará seis caracteres, dando como resultado «JUAN P». Si solamente especificas un número, tal como MID$(A$,4), obtendrás todos los caracteres desde el cuarto en adelante.

El siguiente programa utiliza las funciones LEFT$, MID$ y LEN. Se trata de un juego de anagramas para dos personas. Una de ellas introduce



una palabra que el ordenador seguidamente cifra y presenta en forma de anagrama para que la otra persona la adivine.

```
10 R=RND(-TIME):PRINT
   CHR$(12);"PROGRAMA DE LA
   ADIVINANZA"
20 PRINT"ESCRIBE LA PALABRA
   A ENCONTRAR"
30 W$=""
40 W1$=INKEY$:IF W1$=""
   THEN 40
50 IF W1$=CHR$(13) THEN 80
60 W$=W$+W1$
70 GOTO 40
80 WO$=W$
90 FOR N=LEN(W$) TO 1 STEP-1
100 M=INT(RND(1)*N)+1
110 A$=A$+MID$(W$,M,1)
120 W$=LEFT$(W$,M-1)+RIGHT$
    (W$,LEN(W$)-M)
130 NEXT N
140 PRINT CHR$(12);
    "EL ANAGRAMA ES: ";A$
150 PRINT"CUAL PIENSAS QUE ES
    LA PALABRA"
160 INPUT GU$
170 G=G+1
180 IF GU$<>WO$ THEN PRINT
    "ERRONEA, INTENTA OTRA
    VEZ":GOTO 160
190 PRINT"BIEN HECHO"
200 IF G=1 THEN PRINT
    "NECESITASTE UN INTENTO":
    GOTO 220
210 PRINT"NECESITASTE";G;
    " INTENTOS"
220 PRINT"OTRA PARTIDA
    (S/N)?"
230 A$=INKEY$:IF A$<>"S"
    AND A$<>"N" THEN 230
240 IF A$="S" THEN RUN
```



Cuando ejecutes este juego, observarás que la primera palabra tecleada no aparece en la pantalla. Lo hemos programado así para que tu oponente no pueda saber cuál es. Cada ordenador utiliza un método diferente para conseguir esto. Aquí hemos hecho uso del INKEY$ de la línea 40. Cada letra que tecleas se va almacenando en la variable W$ (línea 60) hasta que pulses RETURN (línea 50).

La rutina de cifrado está contenida en las líneas 90 a 130. Lo que hace esta rutina es extraer caracteres de forma aleatoria de la palabra (W$) y ponerlos todos juntos en A$ que se convierte así en un anagrama.

En la línea 100 se elige un número aleatorio M comprendido entre 1 y la longitud de la palabra, y a continuación la línea 110 extrae la letra M-sima y se la añade a A$. La línea 120 borra este carácter de la palabra original, suprimiendo su parte izquierda hasta el número M y añadiendo la parte restante que queda a la derecha de M. La palabra resultante tiene ahora un carácter menos, pero la vez siguiente, la variable N también es un carácter menor, por lo que el número aleatorio M está otra vez restringido a la longitud de palabra.

Llegará un momento en que se hayan extraído todos los caracteres y el anagrama quedará impreso en la pantalla, momento en que el programa le preguntará a tu oponente cuál era la palabra original. Cuando teclees en la sentencia INPUT la palabra correcta, el programa te responde cuántos intentos has necesitado y le brinda una nueva oportunidad.

Sería muy sencillo modificar este programa para que se puedan leer las palabras a través de una serie de sentencias DATA en vez de utilizar separadamente un INPUT cada vez. También puedes incluir un sistema de puntuaciones que, por ejemplo, asigne diez puntos cada vez que se acierte a la primera, nueve puntos a la segunda y así sucesivamente.

Otro posible uso de este método de troceado es la manipulación de fechas. Incluso en los casos en que las fechas se introducen en forma de números, por ejemplo 27/03/51, en vez del 27 de marzo de 1951. Después de todo, puedes manipular una fecha expresada de esta forma con las leyes normales de las matemáticas. También puedes procesar diferentes partes de una fecha por medio de la aritmética. Puedes calcular por ejemplo la edad de una determinada persona tecleando su fecha de nacimiento, o calcular el número de días transcurridos entre dos fechas dadas; no tienes más que manejar separadamente la parte de los años, la de los meses y la de los días.

Las funciones LEFT$, RIGHT$ y MID$, te permiten separar con facilidad la parte de los días, la de los meses y la de los años de una determinada fecha. Si ahora utilizas las funciones VAL, obtendrás a partir de la cadena de caracteres resultante un número con el que puedes hacer sumas, restas multiplicaciones y divisiones.

LONGITUD DE LAS CADENAS

A veces interesa conocer la longitud de una cadena. Si por ejemplo sólo dispones de una limitada capacidad de

memoria para cada elemento de información introducido desde el teclado, o sólo un espacio limitado de pantalla para presentar dicha información, puede ser de gran ayuda comprobar la longitud de la cadena antes de seguir con el programa.

La función LEN(A$) te da como resultado el número de caracteres que contiene la cadena A$. Se trata de una función numérica y no de una cadena de caracteres, por lo que puedes manipularla con arreglo a las leyes normales del álgebra. Por ejemplo, si A$ = «Sr Juan Perez», LEN(A$) = 13. Pero supongamos que te piden que teclees un nombre para un determinado fichero cuyo formato tabulado para representación en pantalla sólo tiene sitio para 11 letras, por lo que puede ser que haya que enviar un mensaje para que se recorte la entrada, tal como el siguiente:

```
10 PRINT"ESCRIBE EL NOMBRE"
20 INPUT A$
30 IF LEN(A$)>11 THEN PRINT
   "LO SIENTO. SOLO ADMITO
   11 CARACTERES":GOTO 10
```

El usuario podría así recortar su entrada y teclear únicamente «JUAN PEREZ».

En otros casos podría resultar más fácil truncar automáticamente la entrada. Esto se hace con una línea tal como la siguiente:

```
IF LEN (A$>15 THEN A($)=LEFT$(A$,15)
```

CONVERSION DE CADENAS A NUMEROS

Uno de los usos de las variables de cadenas de caracteres es conseguir que las entradas introducidas por el teclado sean correctas. Por ejemplo, si escribes un programa que contenga un par de líneas como las siguientes:

```
100 PRINT"ESCRIBE UN
    NUMERO"
110 INPUT A
```

el ordenador se quedará esperando que teclees algún número. Si por error tecleas algún carácter no numérico, casi todos los ordenadores interrumpirán la representación en pantalla y enviarán un mensaje etándar de error. Dicho mensaje te informará de que hay algo que está mal pero no te dirá lo que es.

Pero si en lugar de esto utilizas:

```
110 INPUT A$
```

en este caso el usuario puede teclear casi cualquier cosa y el ordenador lo aceptará. El siguiente paso es convertir la cadena de caracteres en un número. Para hacerlo, utiliza la función VAL(A$).

La función VAL extrae la parte numérica de la cadena de caracteres. Así, si A$ es un número, entonces VAL(A$) será un número que puede ser utilizado en el resto del programa. Si por el contrario A$ no es un número, VAL(A$) dará como resultado el valor 0. Puedes escribir entonces una pequeña subrutina que explique en detalle lo que el usuario ha hecho mal y le brinde otra oportunidad sin tener que salir del programa y destruir la presentación que haya en pantalla.

Un punto importante a tener en cuenta a propósito de la función VAL es que sólo extrae números del principio de la cadena. En consecuencia, VAL(«25 DE JULIO») será igual a 25, mientras que VAL(JULIO 25) será igual a cero. De modo que ten cuidado al utilizarla.

La función VAL resulta muy útil para ordenar los números de una cadena de caracteres de varias formas diferentes. Si por alguna razón tienes los nombres y las puntuaciones de un grupo de alumnos en forma de cadenas, tal como: A$ = «32 JORGE», B$ = «45 MIGUEL», C$ = «41 ELENA», etc y deseas calcular la media de la clase, puedes extraer valores individuales con ayuda de la función VAL. Así VAL(A$) vale 32, VAL(B$) vale 45 y VAL(C$) vale 41.

También puedes usar VAL de forma parecida para ignorar las unidades que pueden ir acompañando magnitudes tales como pesos y longitudes. El siguiente programa te permitirá apreciar la diferencia:

ORDENA TUS CADENAS

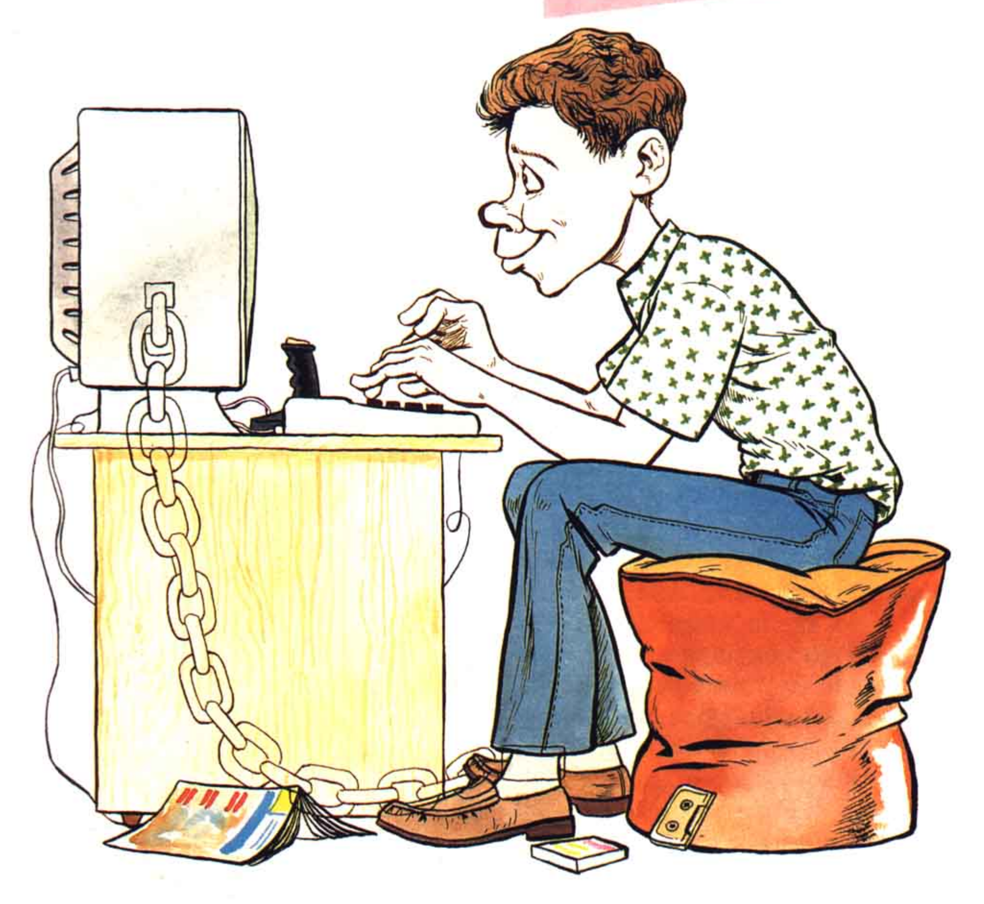
Cuando el ordenador extrae de manera casi mágica las direcciones de centenares de entradas separadas y las presenta en la pantalla, es fácil imaginar que la máquina es más inteligente de lo que realmente es. Sucede casi como si el ordenador estuviera leyendo directamente las entradas y a continuación decidiendo lo que tiene que hacer.

No te confundas. El ordenador se limita a extraer una parte de la cadena a medida que va recibiendo información. Si resulta que una parte de esas cadenas contiene un galimatías, pues eso es lo que te encontrarás. Para asegurarte de que obtienes la misma información de cada cadena, tienes que garantizar que la información se almacena en el mismo sitio. Los investigadores profesionales con frecuencia utilizan tarjetas de entrada con un formato normalizado, que constituyen una de las ayudas más valiosas que te puedes procurar. Para ello no necesitas más que una serie de espacios que contengan los caracteres, rotulados con arreglo al lugar donde debe empezar y acabar cada elemento de información.

```
100 A$="32 KG"
110 B$="110 KG"
120 PRINT"A$+B$=";A$+B$
130 PRINT"VAL(A$)+VAL(B$)=";
    VAL(A$)+VAL(B$)
140 END
```

CONVERSION DE NUMEROS A CADENAS

La función STR$ tiene un resultado casi opuesto al de VAL. Sirve para transformar una cadena en un número. La ventaja de esto es que las cadenas de caracteres pueden ser manipuladas de forma distinta a los núme-



ros, por ejemplo con ayuda del troceado y la concatenación que ya hemos visto, por lo que STR$ tiene varias aplicaciones.

El siguiente programa sirve para convertir en binario un número decimal.

```
10 PRINT"DECIMAL A BINARIO"
20 INPUT"ESCRIBE UN NUMERO
   ENTERO DECIMAL";D
40 B$=""
50 B$=RIGHT$(STR$(D-INT
   (D/2)*2),1)+B$
60 D=INT(D/2)
70 IF D<>0 THEN 50
80 PRINT"EL NUMERO
   BINARIO ES ";B$
```

Evidentemente esto sería más fácil de hacer con la función incorporada BIN$(X) que convierte a binario el número decimal X.

Al introducir un número decimal positivo, la línea 40 hace la cadena B$ igual a la cadena nula, que después va llenando progresivamente con dígitos a medida que progresa el cálculo en el bucle de las líneas 50 a 80.

La línea 50 es la que realmente se encarga de construir el número binario. Para ello resta dos veces la parte entera de la mitad del número decimal del propio número decimal. Esta es una forma sencilla de comprobar si un número es par o impar. Si es impar, el resultado es 1, mientras que siendo par el resultado es 0. Naturalmente, estos son los dígitos binarios buscados.

PROCESAMIENTO DE TEXTOS

Las funciones de manejo de cadenas de caracteres se utilizan ampliamente, en especial en proceso de textos. Una necesidad que se plantea con frecuencia es la sustitución de una palabra por otra en toda la extensión de un documento (posiblemente debido a que has descubierto una falta de ortografía). Naturalmente, si la nueva palabra tiene una longitud diferente de la original, tienes que desplazar el resto del texto para dejarle sitio. El siguiente programa te muestra la manera de hacerlo:

```
10 INPUT"ESCRIBE EL TEXTO"
   ;T$
20 INPUT"PALABRA A REEMPLAZAR
   ";P$
30 INPUT"PALABRA SUSTITUTA"
   ;NP$
35 P=0
40 P=P+1
50 A$=MID$(T$,P,LEN(P$))
60 IF A$<>P$ THEN 90
70 T$=LEFT$(T$,P-1)+NP$+
   RIGHT$(T$,LEN(T$)-P-LEN
   (P$)+1)
80 P=P+LEN(NP$)-1
90 IF P<LEN(T$) THEN
   GOTO 40
100 PRINT T$
110 GOTO 20
```

Empieza introduciendo una frase más bien sencilla y a continuación ensaya el efecto de sustituir algunas de sus letras o palabras. En las palabras cortas tales como «a» o «un» conviene teclear además un espacio antes y después para evitar que se apliquen también los cambios cada vez que se presenten dichas palabras embebidas en otras más largas, tales como «para» o «unidad».

El programa funciona de la siguiente manera: La línea 40 envía a P a empezar la fase de búsqueda al principio del texto. Las líneas 50 y 60 detectan la primera aparición de la palabra o grupo de letras que pretendes sustituir y a continuación la línea 70 realiza la sustitución. Lo que hace es tomar el texto original, remplazar la palabra antigua por la nueva y añadir el texto restante. El proceso se repite hasta que se ha realizado la sustitución en todos los sitios en que aparece la palabra que se quiere sustituir, imprimiéndose a continuación en la línea 100.

Como has visto, se trata de un ejemplo muy simple, los procesadores del mundo real son mucho más complicados. Sin embargo puede servir para darte una idea de algunas aplicaciones prácticas de lo que se puede hacer con las cadenas de caracteres.

UN PROGRAMA QUE JUEGA A LAS CUATRO EN RAYA (II)

En este número publicamos la segunda parte del programa «Cuatro en Raya», junto con un desglosado exhaustivo de sus principales rutinas. De paso, con este programa trataremos de ilustrar las dificultades que presenta el BASIC en esta materia, y sobre todo, cómo superarlas.

En el último artículo publicado dentro de esta serie, ilustrábamos los principios del funcionamiento de los juegos de estrategia con el ejemplo de las «Cuatro en Raya». Creemos que las explicaciones eran lo suficientemente detalladas como para que algún lector avezado se atreviera a desarrollar su propio juego inteligente. Tanto para los que lo han intentado como para los que no, este mes proponemos lo que podría ser la solución al reto que planteábamos en el último número.

Por si algún lector no conociera este juego, o tuviera alguna duda por su similitud con el de las «Tres en Raya», empezaremos por explicar someramente en qué consiste. Se trata de colocar cuatro fichas de un mismo color en línea, ya sea horizontal, vertical o diagonal. Cada ficha ha de colocarse sobre la base del tablero o encima de otra ficha ya colocada, aunque sea del contrario. El primer jugador que consiga su objetivo, será el ganador. Es fundamental tener en cuenta la importancia del hecho de «colocar primero» en el desarrollo posterior del juego. Por esta razón, el programa incopora la posibilidad de elegir quien comienza el juego, el ordenador o el usuario.

Volvemos a hacer hincapié en cómo se demuestra que la **Inteligencia Artificial** puede hacer cosas sorprendentes con listados sorprendentemente breves.

DESGLOSADO DEL PROGRAMA

Para una clara explicación del funcionamiento del programa, nos remitiremos constantemente al listado del mismo y a los gráficos ilustrativos que acompañan al texto.

Como vimos en números pasados, el fundamento básico de los juegos de estrategia está en la valoración de las jugadas posibles, propias y del contrario, para determinar cuál es la más favorable y actuar en consecuencia. Pués bien, en este caso, como en todos, deberemos empezar por establecer cuáles son las posiciones en las que podemos colocar una ficha. La figura número uno lo muestra (conociendo las reglas del juego, no necesita más comentario). Nosotros podemos verlo a simple vista, pero ¿cómo hacerselo ver al ordenador?

De esto se encarga la rutina-bucle situada en la línea 1030-1060.

El tablero está representado en la matriz a(7,4), con un cero para las posiciones vacías, un uno para las ocupadas por una ficha del ordenador, y un dos para las ocupadas por el jugador. El bucle 1030-1060 comienza a buscar posiciones vacías desde la base del tablero, enviando el flujo del programa a la parte siguiente (valoración de la posición) cuando encuentre una. Para ver la relación tablero/matriz (que está pensada a la inversa de como cabría esperar para simplificar el proceso), consultar la figura cuatro.

A partir de aquí, posición por posición, el programa estudia y valora los alineamientos conseguidos. Almacena el valor más alto conseguido en la variable ta, opera de la misma manera con las posibles jugadas del contrario, y finalmente hace la jugada más favorable. Si la mejor jugada del ordenador es un alineamiento de dos fichas, y la mejor jugada posible del contrario otro de tres, el ordenador hará la

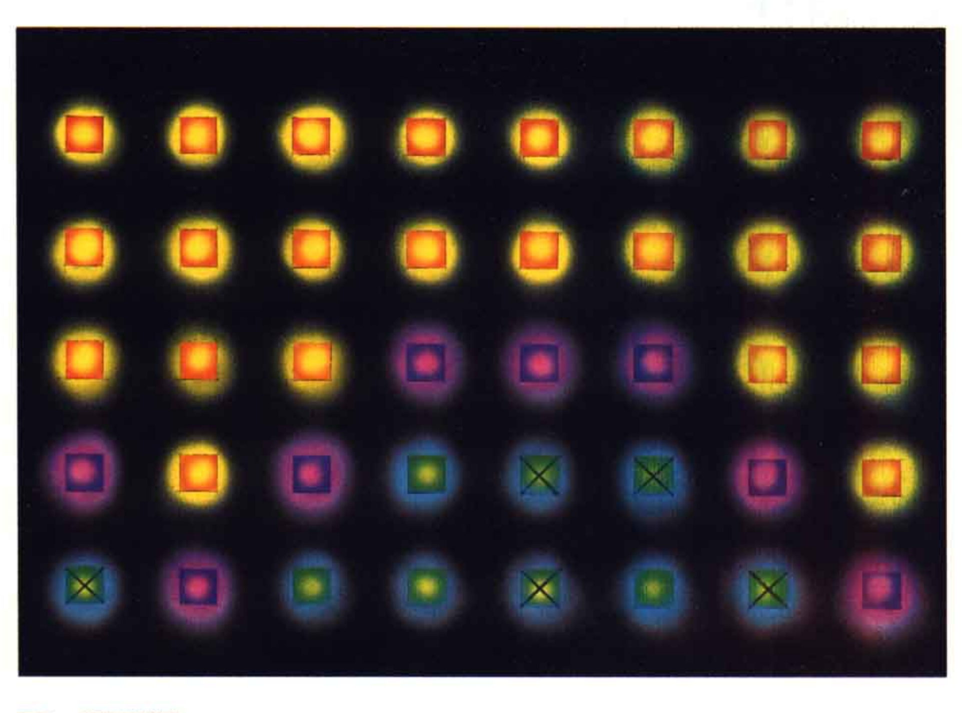
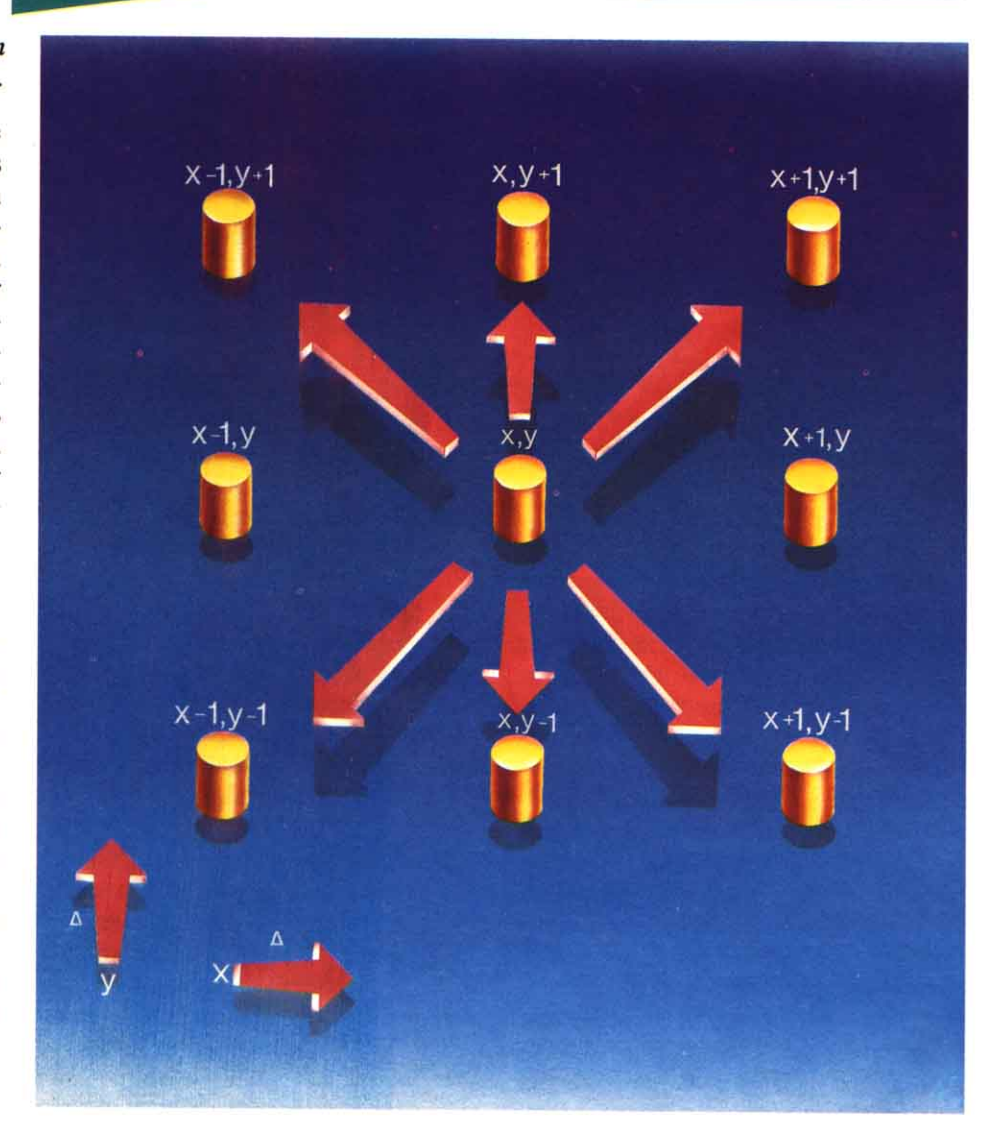


Figura 1.- Los lugares a analizar en la presente situación aparecen coloreados en magenta.

Figura 2.- Análisis de una posición con respecto a las posiciones contiguas.



jugada del contrario para evitar que éste la realice, pues su valor (tres) es mayor. El programa puntúa con un punto cada ficha alineada en una posición dada. Este análisis y valoración configura lo que podemos considerar como la parte más importante del programa, que se encuentra entre las líneas 1100-1530. En cada posición analizada, se estudian las alineaciones conseguidas en la forma que muestra la figura número dos. Se comienza por analizar una posición contigua; por ejemplo, x+1,y−1. Si está ocupada por una ficha del mismo color (representada en la matriz por un 1 si es del ordenador, y por un 2 si es del contrario), se suma un punto al valor de esa posición, que comenzó con cero, y se vuelve a sumar uno a 'x' y a restar uno a 'y' para seguir la línea en esa dirección, es decir, se analiza la posición x+2, y−2. Si no está ocupada por una ficha del mismo color, se vuelve a la posición original (x,y), y se pasa a seguir en el sentido contrario dentro de la misma línea (es decir, x−1, y+1), procediendo de la misma manera. Al final obtendremos el número de fichas alineadas y contiguas en la recta representada por las dos flechas verdes. Sucesivamente, se hace un análisis idéntico con el resto de las rectas (la horizontal, vertical y la otra oblicua) y se almacena, como ya dijimos, el mayor alineamiento en la variable ta.

Quizás esta explicación resulte un poco complicada si se hace con palabras. La mejor forma de comprender esto que hemos expuesto es seguir detenidamente el listado del programa, para estudiar directamente la forma en que lo hace.

Vamos a tomar ahora el ejemplo representado en la figura tres. El ordenador estudia y valora dos jugadas propias posibles, que se realizarían colocando en la posición 1 o en la posición 2. En la posición uno, obtendría un alineamiento de dos fichas, con un valor pues de 1 punto (la ficha que aún no se ha colocado no se cuenta, por eso es uno y no dos), mientras que en la posición 2, el alineamiento sería de 3 fichas, con un valor de 2. La posición uno, por tener un valor menor, queda descartada. A continuación, se busca la mejor jugada del contrario (impropiamente, decimos «mejor jugada» cuando deberíamos decir «alineamiento más numeroso», ya que en realidad, un alineamiento de tres no tienen por qué ser necesariamente más beneficioso que uno de dos en la estrategia general del jugador).

Colocando éste en la posición tres, obtendría un alineamiento de cuatro fichas, y por tanto vencería. En todo caso, como su valor es mayor que el obtenido por el mejor alineamiento del ordenador, el programa pondrá su ficha en la posición tres para «taponar» la jugada victoriosa del contrario.

Los lectores más atentos habrán observado que en el programa no se emplean otros criterios de valoración de los que sí hablamos en el número anterior, como por ejemplo la posibilidad de seguir colocando fichas a ambos lados del alineamiento. Lo hemos hecho así por dos razones fundamentales: en esta serie, como ya hemos advertido, los programas que proponemos no pretenden ser pautas rígidas y acabadas en todos sus detalles, sino materiales para que el lector trabaje con ellos y desarrolle sus propias mejoras y creaciones; este hecho nos obliga también a no tratar de complicar excesivamente el listado, para hacerlo accesible sin necesidad de dedicar largas horas a su estudio.

No obstante, podemos proponer una mejora que permita valorar especialmente las posiciones situadas en el centro del tablero, por ofrecer más po-

sibilidades de completar una línea. Bastaría simplemente con sumar un punto al valor almacenado por cada combinación si en esa posición las variables n y f son distintas de cero, de cuatro y de siete, es decir, si esa posición no está en un borde del tablero.

Entre las líneas 2000-2130, el programa trabaja todo lo relacionado con el turno del jugador. Vigila que éste no trate de colocar en un lugar en que no pueda hacerlo, mueve el cursor de una posición a otra e imprime la ficha en el lugar elegido.

El cursor se desplaza con las teclas o,p,q,a, y la ficha se coloca en el lugar elegido pulsando «cero». Cuando al principio del juego, el programa nos pregunta si queremos que empiece él, basta con pulsar la «s» para contestar que si, y la «n» para no. En este último caso, comenzaríamos nosotros.

A pesar de la sencillez del algoritmo empleado en este programa, observaremos que resulta difícil vencerlo, sobre todo si empieza él a jugar. Sabiendo cómo funciona y en qué se basa su análisis, podremos encontrar un sistema para ganar siempre, pero también otro sistema para que la máquina lo evite.

Finalmente, hemos de advertir que la introducción de mejoras en el programa que supongan un aumento del listado, sobre todo en la parte correspondiente al turno del ordenador, puede suponer un considerable incremento de la lentitud de la máquina en la resolución de sus jugadas.

LA RUTINA 2040

Los «devoradores» de bibliografía y revistas de informática habrán podido comprobar con cuánta frecuencia se invocan las «limitaciones del BASIC» para disculpar a lo que en algunas ocasiones no es más que el desconocimiento de las posibilidades de este lenguaje de programación. La línea 2040 del programa que acompaña a este texto, encargada de determinar qué tecla se ha pulsado y variar el valor de la posición en que se ha de imprimir el cursor, es un claro ejemplo de un correcto aprovechamiento del BASIC, en aras a una mayor velocidad y un menor gasto de memoria.

Las condicionales del tipo

IF...THEN...

son extraordinariamente lentas. Varias de ellas en sucesión colocadas en líneas distintas, pueden inutilizar un programa al hacerlo excesivamente lento. Pero como las sentencias condicionales resultan imprescindibles para casi todo lo que queremos hacer, se hace necesario idear otro tipo que consuma menos memoria y menos tiempo, como el que proponemos en la línea 2040.

Las expresiones colocadas entre paréntesis en el ejemplo que ponemos, dan siempre como resultado uno o cero, según sean verdaderas o falsas. Como los valores que tenemos que sumar o restar al valor de las variables que almacenan la posición del cursor, n y f, son siempre uno o cero, podemos utilizar las posibilidades de la sentencia AND a través de una serie de operaciones lógicas para alterar esas variables directamente.

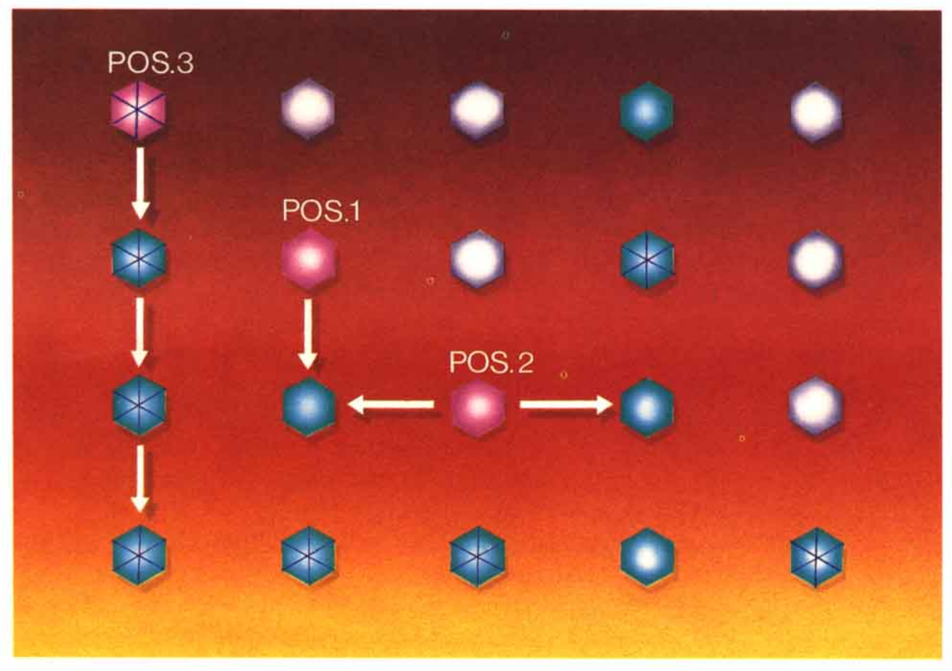
Veamos ahora lo que hubiéramos necesitado hacer para conseguir el mismo efecto con sentencias del tipo IF ... THEN...

```
2040 IF i$=8 AND n<8 THEN LET n=n+1
2042 IF i$=5 AND n>1 THEN LET n=n−1
2044 IF i$=7 AND f<5 THEN LET f=f+1
2046 IF i$=6 AND f>1 THEN LET f=f−1
```

La comparación, como se ve, no deja lugar a dudas.

Esperamos que el lector pase un buen rato con el programa «CUATRO R.», y que no se limite a teclearlo y dejarlo tal cual está.

```
2000 REM TURNO DEL JUGADOR
2005 IF T=1 GOTO 1000
2010 N=1: F=1
2015 LOCATE 0,1: PRINT
     "SU TURNO          "
2018 GOTO 2060
2020 I$=INKEY$
2025 IF I$="" GOTO 2020
2030 LOCATE X,Y: PRINT " "
2035 IF I$="0" GOTO 2100
2040 N=N+(1 AND I$="P" AND
     N<7)-(1 AND I$="O" AND
     N>0): F=F+(1 AND I$="A"
     AND F<4)-(1 AND I$="Q"
     AND F>0)
2060 X=(N*3)+2: Y=(F*3)+5
2070 LOCATE X,Y: PRINT
     CHR$(&HDB): GOTO 2020
```



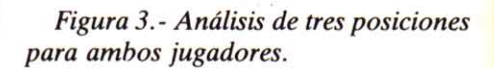

Figura 3.- Análisis de tres posiciones para ambos jugadores.

Figura 4.- Estructura de la matriz a(n,f).

```
2100 REM FINAL TURNO DEL
     JUGADOR COMPROBACION
     VICTORIA E IMPRESION DE
     LA FICHA
2102 IF F=4 GOTO 2108
2105 IF A(N,F+1)=0 GOTO 2060
2108 IF A(N,F)<>0 GOTO 2060
2110 X=(N*3)+2: Y=(F*3)+4
2115 LOCATE X,Y:PRINT CHR$(1)
     ;CHR$(66)
2117 BEEP
2118 A(N,F)=2: T=1: TT=2:
     TA=0: TF=0: U=1: GOSUB
     1100
2119 IF TA<>3 THEN GOTO 1000
2120 IF TA=3 THEN CLS: PRINT
     "ME HAS GANADO": PRINT "
     PULSA UNA TECLA": BEEP
2130 IF INKEY$="" GOTO 2130
2140 RUN
2500 REM TABLAS
2510 CLS: PRINT "HEMOS
     QUEDADO EN TABLAS":PRINT
     "PULSA UNA TECLA"
2520 IF INKEY$="" GOTO 2520
2530 RUN
9000 REM TRAZADO TABLERO.
     ELECCION INICIO PARTIDA
9010 FOR N=4 TO 16 STEP 3
9020 LOCATE 2,N: PRINT "I
     I  I  I  I  I  I
     I"
9030 NEXT N
9040 LOCATE 0,0: PRINT
     "CUATRO EN RAYA. E.V.-
     INPUT"
9050 PRINT "EMPIEZO YO ?"
9060 I$=INKEY$
9065 IF I$="" GOTO 9060
9070 IF I$="S" THEN T=1
9080 IF I$="N" THEN T=2
9100 GOTO 1000
```

¡EN CONTRA DE LO QUE DIGA LA GENTE, YO CREO QUE EL ENTENDIMIENTO ENTRE PADRES E HIJOS ES POSIBLE!

TODO ES CUESTIÓN DE QUE HAYA INTENCIÓN DE DIÁLOGO, QUE CADA PARTE CEDA AUNQUE SEA UN POQUITO, QUE INTENTEMOS ACERCARNOS...







LOS MEJORES DE INPUT MSX

PUESTO	TITULO	PORCENTAJE
1.º	Soccer	20 %
2.º	H.E.R.O.	14,1 %
3.º	Knight Lore	13,5 %
4.º	Profanation	9,6 %
5.º	Alien 8	9,2 %
6.º	Yie Ar Kung Fu	8,3 %
7.º	Hyper Rally	7,7 %
8.º	BC II Grog's Revenge	7,3 %
9.º	Bounder	5,2 %
10.º	Yie Ar Kung Fu	5,1 %
		100 %

Para la confección de esta relación únicamente se han tenido en cuenta las votaciones enviadas por nuestros lectores de acuerdo con la sección «Los Mejores de Input».

Diciembre de 1986.

Ordena tus propias ideas

Le sacarás partido a tu ordenador



MSX:
Guía del programador y manual de referencia
T. Sato, P. Mapstone e I. Muriel
2.279 ptas.



LENGUAJE MAQUINA MSX.
Introducción y conceptos avanzados
Joe Pritchard
1.537 ptas.



EL LIBRO GIGANTE DE LOS JUEGOS PARA MSX
Andrew Lacey
1.590 ptas.



DESCUBRE TU MSX.
Programación y aplicaciones
Joe Pritchard
1.272 ptas.

INTELIGENCIA ARTIFICIAL
Conceptos y programas
Tim Hartnell
1.484 ptas.

EL SUPERLIBRO DE LOS JUEGOS PARA ORDENADOR
Tim Hartnell
2.120 ptas.

CODIGOS Y CLAVES SECRETAS
Criptografía en Basic
Gareth Greenwood
1.378 ptas.

PROGRAMACION DEL Z80
Rodnay Zaks
2.915 ptas.

SISTEMAS EXPERTOS
Introducción al diseño y aplicaciones
Tim Hartnell
2.120 ptas.

SIMULACIONES
Replica la realidad con tu ordenador
Tim Hartnell
1.643 ptas.

☐ Les ruego me envíen el catálogo de su editorial.
☐ Les ruego me envíen los siguientes títulos:

____

____

TOTAL __________

☐ Adjunto talón bancario a GRUPO DISTRIBUIDOR EDITORIAL, S. A.

☐ Pagaré contrarrembolso (+ 140 pesetas de gasto de envío).

Nombre ______________________________

Profesión ______________________________

Dirección ______________________________

C. P. __________ Localidad __________

Provincia ______________________________

IM



Adquiéralos en su librería habitual.
Si no le es posible o desea que le enviemos nuestro catálogo, envíe este cupón a:
Apdo. de Correos 14632, Ref. D. de C. 28080 MADRID
Comercializa: GRUPO DISTRIBUIDOR EDITORIAL.

ANAYA ANAYA ANAYA ANAYA ANAYA ANAYA

EL LAGO ENVENENADO

Drixx se encuentra perdido en un mundo desconocido para él. Cuando su platillo volante falto ya de combustible pierde altura peligrosamente, **Drixx** sólo puede aterrizar en medio de un venenoso lago de cloro. Se encuentra atestado de innumerables peligros que nuestro héroe deberá esquivar. Temibles rayos, objetos agresivos, compuertas que se abren y se cierran intentando atraparle, y un sinfin de peligros más que **Drixx** sólo puede intentar esquivar o saltar sobre ellos. Pero ese salto no está exento de peligros, ya que si cae al lago, su suerte estará echada.



No dudamos de que con tu pericia, **Drixx** conseguirá llegar sano y salvo a la orilla.
The Wall es un juego que desarrolla una buena idea, pero cuya realización, si tenemos en cuenta el nivel de sofisticación alcanzado por otros juegos actualmente, se ha quedado un poco pobre, tanto a nivel de gráficos como de animación, de variedad de efectos y de sonido.

Es un juego al que se le podía haber sacado mucho más partido.



DATOS GENERALES

TITULO The Wall

FABRICANTE Erbe

CLASE DE PROGRAMA Juego

FORMATO Cassette

CALIFICACION (Sobre 10 ptos.)

ORIGINALIDAD	7
INTERES	7
GRAFICOS	6
COLOR	8
SONIDO	6
TOTAL	34





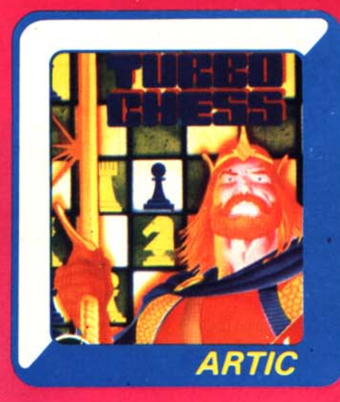
★★★★★★★★★★★★★★★★★★★★★★★

UN POKER MUY SEXY

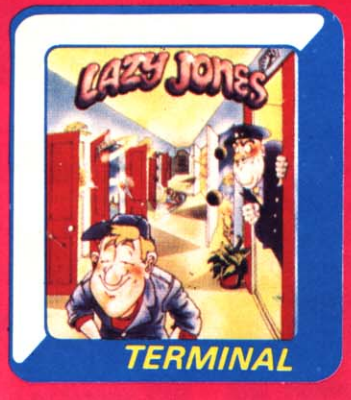
¿Habéis probado alguna vez la emoción del poker descubierto? Ahora tenéis ocasión de disfrutarlo con las dos interesantes modalidades de este programa.
En la primera de ellas nos enfrentaremos en una interesante partida de *strip-poker* con **Samantha Fox**. Ella está convencida de que somos unos malos jugadores y ha aceptado apostar sus prendas de vestir.
No va a ser fácil, sin embargo, llegar a «derrotarla» completamente, ya que **Samantha** es una experta jugadora, habilísima con los trucos y las tretas del juego, y que defenderá con uñas y dientes cada partida ¡Cuidado, no sea que tengamos que desnudarnos nosotros al final! Os garantizamos que no es nada sencillo derrotarla.
En la segunda cara del *cassette* está la segunda modalidad de este juego.



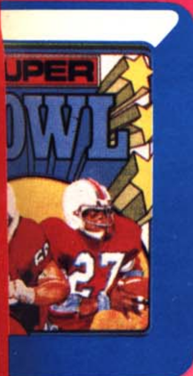
KNIGHTS

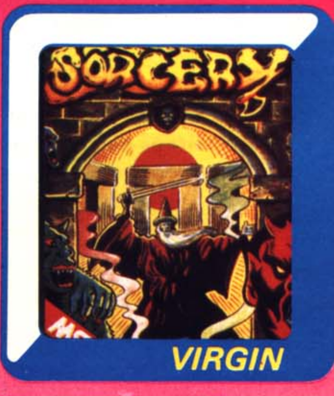


ARTIC

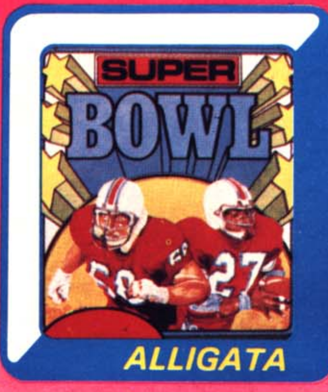


TERMINAL





VIRGIN

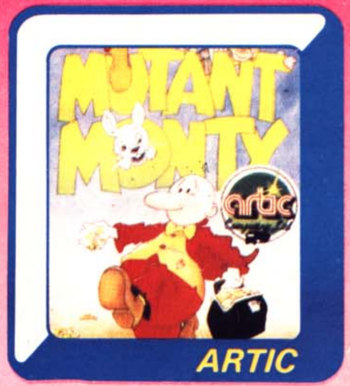


ALLIGATA





ALLIGATA



ARTIC



MIRRORSOFT





ALLIGATA



TYNESOFT







- **THE KNIGHT COMMANDER:** 40 nuevos comandos para el Basic del MSX dejando completamente libre la memoria RAM. 22 Teclas extras. RS 101
- **TURBO CHESS:** Completísimo juego de Ajedrez. 100% código máquina. Efectos sonoros. Niveles de dificultad. RS 108
- **LAZY JONES:** Una serie de juegos en uno solo. Sprites en alta resolución. Niveles progresivos de dificultad. Efectos sonoros. RS 106
- **SORCERY:** Esta emocionante arcade aventura te transporta a un mundo fantástico. 50 pantallas para explorar. Detallados gráficos. RS 103
- **SUPER BOWL:** Una cuidada y realista simulación del fútbol americano. Uno o varios jugadores. RS 104
- **3D KNOCKOUT:** Un combate de boxeo con ocho niveles de dificultad y ocho oponentes. RS 102
- **MUTANT MONTY:** 40 pantallas. Joystick o teclado. Sprites. 100% código máquina. Efectos sonoros. RS 107
- **STAR SEEKER:** Los misterios del Firmamento son revelados en este programa. Una introducción facilmente comprensiva a la fascinante ciencia de la Astronomía. RS 105
- **WHO DARES WINS II:** Conviértete en un voluntario para una misión suicida. Un juego rebosante de acción. RS 111
- **WINTER OLYMPICS:** Una cuidada simulación de las más importantes pruebas olímpicas de los juegos de invierno. RS 109

✱IMPORTACION:

-ALINEADOR DE CABEZAS
GS 102

¡¡APARICION INMEDIATA!!
EUROPEAN GAMES.

DISTRIBUIDO POR:



DISCOVERY informatic
Arco Iris, 75
T. 256 49 08-09
08032-BARCELONA

Consiste en una variante del poker descubierto con 7 cartas, que podemos jugar contra uno, dos o tres jugadores. El ordenador hace de árbrito, y nos indicará cada una de las jugadas que podemos realizar, tales como pasar, plantarse, apostar, envidar y subir.
La partida se desarrolla en un ambiente de tensión e intriga ya que el juego es con dos cartas cubiertas y el resto descubierto, de forma que cada uno de los jugadores deberá tratar de adivinar el juego de los demás por sus cartas y su comportamiento (¡suponiendo que no vayan de farol, claro!).
El ordenador analizará nuestro juego y se encargará también de guiar el comportamiento de los otros tres jugadores como si fuesen rivales entre sí, lográndose un sorprendente efecto de realismo y emoción.

DATOS GENERALES

TITULO Samantha Fox. Strip-Poker

FABRICANTE Martech

CLASE DE PROGRAMA Juego

FORMATO Cassette

CALIFICACION (Sobre 10 ptos.)

ORIGINALIDAD	8
INTERES	9
GRAFICOS	8
COLOR	7
SONIDO	7
TOTAL	39



El nivel de habilidad aunque se ha dotado al programa, y la capacidad de análisis de jugadas y de comportamientos es sorprendente, lo que indica que se han seguido técnicas de inteligencia artificial en su desarrollo. No es fácil ganar a **Sid**, **Pat** y **Joe** así que ¡paciencia y cuidado con perderlo todo en la partida!

★★★★★★★★★★★★★★★★★★★★★★★★★★★★★

BUSCANDO LA SALA CENTRAL

DATOS GENERALES	
TITULO Drome	
FABRICANTE The Bytebusters	
CLASE DE PROGRAMA Juego	
FORMATO Cassette	

CALIFICACION (Sobre 10 ptos.)	
ORIGINALIDAD	9
INTERES	9
GRAFICOS	8
COLOR	8
SONIDO	7
TOTAL	41

La malvada computadora nos tiene atrapados dentro de sus circuitos. Nuestra única salvación es llegar a la sala central y destruir su control principal con nuestros disparos de energía.
Pero antes tendremos que superar las pruebas de las motos de luz, de los tanques amenazadores y del viaje fantástico por un conductor eléctrico. En la primera de ellas, tendremos que conseguir que la moto que guía choque contra su propia estela luminosa. En la segunda, tendremos que guiar nuestro tanque por un complicado laberinto, para destruir los tanques que ella controla. En la tercera viajaremos con nuestra micronave por un conductor eléctrico, viendo los electrones y los protones, que chocarán contra nosotros.

Los electrones no son ningún problema, pero si chocan muchos protones contra nuestra nave, debido a su mayor masa, acabarán destruyéndola.
Al llegar a la sala central de la computadora tendremos que esquivar los relámpagos luminosos que nos lance, y a la menor oportunidad destruir su módulo central para alcanzar nuestra libertad y el mundo exterior.









No es una aventura, son cuatro en un juego. La animación y el color son estupendos, seguro que pasáis ratos muy divertidos con él.

****

RESCATE EN EL MAR DEL NORTE

Una terrible tormenta se ha desencadenado en el Mar del Norte. La plataforma petrolífera está en peligro. Algunos hombres, sin poder resistir el embate de las olas, han caido al agua, y corren grave peligro de ahogarse.
La alarma general suena. Todos los pilotos de los helicópteros de rescate deben alzar el vuelo, a la búsqueda de los hombres perdidos, lo antes posible.
No va a ser tarea fácil, pues la tormenta es muy grande, el mar está embravecido y levanta gigantescas olas que amenazan con atrapar nuestro helicóptero. Además las fuertas rachas de viento hacen peligrar la estabilidad de nuestro helicóptero.
Para volar, disponemos de un avanzado cuadro de mandos y de la ayuda de nuestro super-computador de abordo, que hablará con nosotros, ayudándonos en caso necesario.
Hay que darse prisa pues, si llega la noche, perderemos totalmente la





DATOS GENERALES

TITULO North Sea Helicopter

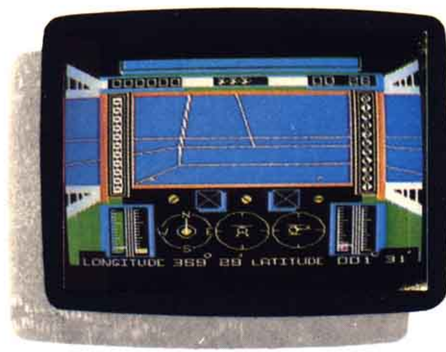
FABRICANTE Aacksoft

CLASE DE PROGRAMA
Rescate aéreo

FORMATO Cassette

CALIFICACION (Sobre 10 ptos.)

ORIGINALIDAD	8
INTERES	8
GRAFICOS	9
COLOR	8
SONIDO	10
TOTAL	43



visibilidad para rescatar a los hombres perdidos y no tendremos más guía que el sensor de infrarrojos.
Es un estupendo simulador de vuelo de helicóptero, tan real, que se maneja exactamente como los helicópteros de verdad, y hay que emplear todos los trucos que los pilotos experimentados utilizan. Por si nosotros, como es lo más probable no lo sabemos, en las instrucciones del juego vienen unas buenas nociones del manejo y funcionamiento de estos aparatos.
Hay que destacar el sintetizador de voz incorporado, que permite al ordenador de la nave ayudarnos indicándonos, por ejemplo, cuando tenemos un hombre bajo nosotros.

★★★★★★★★★★★★★★★★★★★★★★★★★★★★★

EL HOMBRE MURCIELAGO

Cuando juegues con **Batman** no podrás evitar esa sensación de haber jugado a algo parecido en alguna parte... y será cierto, pues **Batman** sigue el esquema de muchos juegos ya aparecidos en el mercado, en los cuales el héroe de la aventura tiene que moverse por un laberinto tridimensional de habitaciones hasta encontrar la forma de salir.
En este caso, el famoso superhéroe se encuentra perdido en un



DATOS GENERALES

TITULO Batman

FABRICANTE Ocean

CLASE DE PROGRAMA
Juego

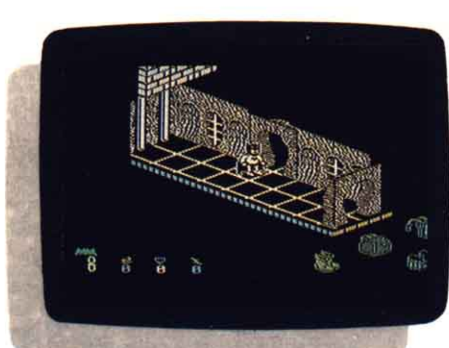
FORMATO Cassette

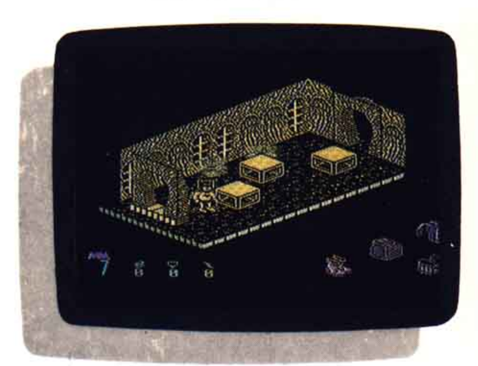
CALIFICACION (Sobre 10 ptos.)

ORIGINALIDAD	6
INTERES	8
GRAFICOS	9
COLOR	7
SONIDO	7
TOTAL	37

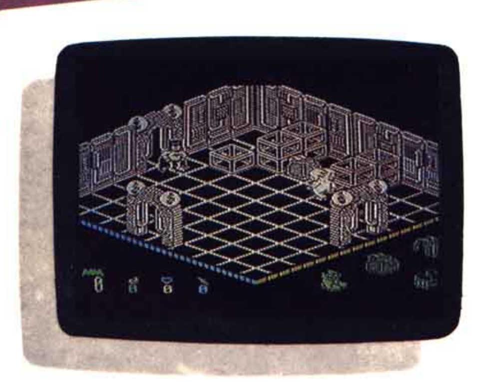








tenebroso castillo. Las fuerzas del mal, han destrozado su nave **Batcsaft**, y han escondido cada una de las partes en una habitación.
El castillo está repleto de horribles criaturas malignas: hombres lobo, *robots* aniquiladores, perros monstruosos y muchos horrores más, que tratarán de destruir al superhéroe y evitar así que



reconstruya su nave para poder escapar.
Además **Batman** no puede confiar en nada, pues zonas que parecen seguras pueden esconder trampas terribles.
Pero no hay que preocuparse. No dudamos que con tu ayuda, el legendario héroe saldrá airoso de todos los peligros que le acechan.

****

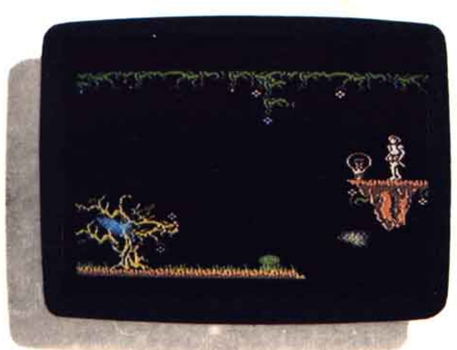
EL GUERRERO DE CAMELOT

Ten cuidado, caballero andante, si osas traspasar la barrera que da paso a los peligros y aventuras de este juego.





En **Camelot** las fuerzas malignas han hecho aparecer cuatro elementos de nuestra era, el siglo XX. Estos elementos se conocen en **Camelot** como la Voz de Otro Mundo, el Espejo de la Sabiduría, el Elixir de la Vida y el Fuego que No Quema.
Estos elementos misteriosos están perturbando el equilibrio de las Fuerzas Mágicas y es tu misión de Caballero Andante entregárselas a los Guardianes de los Mundos para su destrucción.
Pero para eso tienes que enfrentarte con los peligros de cuatro mundos distintos: el Bosque, donde vive **Aznath**, el druida, el Lago, morada de **Kindo**, hermano de **Neptuno** rey del lago, las terribles grutas, donde habita el dragón **Azornic**, señor de los mundos subterráneos, y por último el castillo de **Camelot**, y la **Corte del Rey Arturo**. Allí se encuentra la solución final.





DATOS GENERALES

TITULO Camelot Warriors

FABRICANTE Dinamic

CLASE DE PROGRAMA Juego

FORMATO Cassette

CALIFICACION (Sobre 10 ptos.)

ORIGINALIDAD	8
INTERES	8
GRAFICOS	9
COLOR	9
SONIDO	7
TOTAL	41

Es seguro que con **Camelot Warriors** tenemos muchas horas de diversión aseguradas. Los efectos gráficos son excelentes, así como la originalidad de los efectos y el movimiento. A través de todos los mundos por los que nuestro caballero andante va a emprender su aventura encontramos llamativas plantas, formas y colores insospechados... y también grandes peligros, contra los cuales sólo tenemos nuestra espada y nuestra destreza. ¡Atrévete y lánzate a la aventura!

****



EL ZOCO

Vendo HB-55P junto con ampliación memoria 16K. Precio a convenir. Llamar a:

Victor
Tel. 26 14 29
Murcia

Vendo/Cambio juegos MSX poseo entre otros: Cadoo, Ninja, Karate, Zaxxon, Hero... Interesados llamar o escribir a:

Javier Rosendo López
Avda. Concha Espina s/n
Tel. (942) 70 05 16
39500 Cabezón de la Sal
Cantabria

Intercambio los cartuchos de Konami: Soccer, Tenis y Antartic Adventure poseo hasta 30 juegos y cartuchos. Llamar a:

Juan Manuel
Tel. (954) 66 39 80 (de 19 a 22 h.)

Desearía contactar con usuarios del MSX para poder intercambiar programas (Alien 8, Night Shade, Gunfrigt, Zaxxon, Night Lore, etc).

Angel Ramón Pérez Corzo
García Lorca s/n
Tel. (924) 34 21 28 (tardes desde las 8)
Arroyo de San Servan
Badajoz

Intercambio programas comerciales MSX en cinta o disco de 3.5, primeros títulos del mercado.

Iñaki Fernández Izquierdo
Zamakoa 7, 5° dcha.
Tel. (94) 456 33 72
Galdácano (Vizcaya)

Cambio programas MSX, únicamente con usuarios de mi localidad. Poseo Manic Miner, Jet Set Willy, Sorcery, Superbowl, Road Fighter (en cinta), etc. Llamar o escribir a:

Pablo Pardo Márquez
1° de Mayo, portería 2, 3° 2ª
Tel. 870 91 45
Granollers (Barcelona)

Cambio un organo Casio PT-1. 10 horas de uso y garantía oficial de un año por el Boxing o el Golf de Konami o vendo por 4.000 ptas.

Juan Carlos Sánchez
Illescas, 119, 1°C
Tel. 718 15 89
28024 Madrid

Compro cartucho de expansión de memoria 64K. Precio a convenir.

Ximo Martínez Rojo
Av. Horchata, 31, 6
Tel. (96) 362 71 39
Alboraya (Valencia)

Cambio Ratón MSX y programa Cheese por 150 juegos. También lo compraría.

Juan Manuel
Tel. (954) 66 39 80
Sevilla

Desearía intercambiar programas MSX. Poseo: Knight Lore, Zaxxon, Yie ar Kung-Fu II, River Raid, Bounder, Jet Set Willy II, Chiller, Slapsher (Hockey hielo) etc... Me interesan: Profanation, The Way of the tiger, Hero, Gunfright o Jack the Nipper. Llamad a:

Tel. (93) 379 20 86 (a partir de las 6 de la tarde)

Vendo Mitsubishi 80K modelo ML-FX2 con 4 programas incorporados en el ordenador: Base de Datos, Procesador de Textos, Cálculo electrónico, Creación de Gráficos, con sus respectivos cables y libros en castellano y una cinta de demostración. Regalo 1 cassette con juegos de marca. Precio a convenir. Preguntar por:

Yon
Tel. (943) 39 63 36
San Sebastián

Deseo cambiar ordenador Mitsubishi con 80K RAM y teclado numérico + 2 manuales en español + cassette especial + joystick + revistas y unos 30 juegos (Hero, Profanation, Sorcery, etc...) por Philips, por Hit Bit o por Dynadata, pero que tenga la misma memoria. Interesados llamar a:

Roberto
Tel. (982) 44 00 10
(sólo de 9 a 11 de la noche)

Cambio 4 juegos por uno: Chiller, Space Walk, Misión de Combate y Krypton por uno de estos: Hero, Profanation, Zakil Wood, Alien 8, o Hiper Rally. Poneros en contacto conmigo:

Victor Valero Berenguer
Mayor de Salt, 252, bajos
Tel. (93) 23 40 38 (excepto lunes y domingos)
19170 Salt (Gerona)

Vendo ordenador Yashica 64K sistema MSX, con monitor Philips fósforo verde, cassette Philips tipo consola especial para ordenador, dos joysticks Sony, dos cartuchos, montones de juegos y revistas por sólo 80.000 ptas con menos de tres meses de uso. Más información llamar a:

Carlos
Tel. 858 17 31
Madrid

Vendo Sony HBF 500 P (MSX II) completo, nuevo. Incluidos manuales y Sistema Operativo (diskette MSX-2). Por imposibilidad de envío, sólamente a tratar en cercanías de mi población. Urgente. Precio: 130.000 ptas.

José Luis Basterretxea Barroso
Iturribide, 12, 2° dcha.
Tel. 469 43 63 (llamar de 2 a 4 h.)
Getxo (Bizkaia)

Vendo Sony HB-75 P de 80K más libros más programas comerciales por 30.000 ptas.

Venancio Bidasoro
Harzubia, 9
Tel. 81 25 06 (de 3 en adelante)
Azpeitia (Guipuzcoa)

Vendo ordenador Hit Bit 55P, ampliación 16K, 4 cartuchos y 20 juegos por 40.000 ptas. Interesados llamar a:

Francisco
Tel. 10 51 42 (de 9 a 11 de la noche)
Almeria

Cambio juegos de ordenador, primeros títulos y todas clases, o los vendo. Contactar conmigo en esta dirección y teléfono:

Jordi Pujolá
Calle N-ª 7, n°23
Tel. (93) 664 19 71
Castelldefels
Barcelona

Vendo/Cambio cintas de juegos. Tengo Basket-ball, Zaxxon, Minerva III, Polar Star, y 20 más. También vendo joystick con 2 disparadores y opción rápido-normal. Mandar lista:

Alonso Gómez
La Quintana, 4
Santpeda (Barcelona)

Cambio 35 juegos comerciales, 15 de ellos de Konami por cartuchos de juego, interesados mandar el nombre del cartucho. Escribir a:

Juan José Ritoré
García de Paredes, 19
Tel. 23 76 87
Badajoz

Intercambio todo tipo de programas MSX con chicos y chicas de toda España. Poseo más de 50 títulos.

EL ZOCO

Vicente Villar Elizondo
La Paz, 34, 2°D
Baracaldo (Vizcaya)

Intercambio juegos MSX. Poseo: Jack the Nipper, Gunfright, Knight Lore, Hero, Ghostbusters y muchos más.

Ladislao Banfalvi
Plaza Mayor, 23
Tel. (965) 85 26 75
Benidorm (Alicante)

Cambio ampliación 32K + 6.000 por ampliación 64K en buen estado.

Daniel Moreno Garriga
General Prim, 7, 3° 2ª
Tel. (93) 873 63 28 ó 873 63 90
Manresa (Barcelona)

Vendo ampliación de 16K Sony HBM-16 por 3.500 ptas. en perfecto estado. Preguntar por:

José Ignacio
Tel. (94) 449 46 85

Necesito manual o copia en castellano de impresora Seikosha GP-500 AS. Acepto donación de ordenador o impresora que no uses por cualquier motivo. Abono partes.

Julián Seguen
Serradilla, 28
Tel. 705 98 24
28044 Madrid

Intercambio programas e información en MSX. Dispongo de los mejores: Gunfright, Knight Lore, Road Fighter, Alien 8, Hero... Me interesan sobre todo programas de aplicación tales como: ensambladores, desensambladores, procesadores de texto, base de datos, etc. Busco compañero iniciado para concurso Sony. Interesados dirigirse a:

Juan Luis Garrido Castro
Dulcinea, 32
Tel. 25 14 97
14014 Córdoba

Vendo videojuegos Atari, joysticks diversos y 19 cartuchos por sólo 2.500 ptas.

Ignacio Pérez Cuadrado
Tolosa Latour, 4, 6°C
Cartagena (Murcia)

Intercambio, vendo, compro juegos en cinta de MSX. Tengo entre otros: Knight Lore, Night Shade, Hero, River Raid, Yier Ar Kung-fu, Boxing, Ghostbuster, Profanation, etc. Desearía tener: Green Beret, Sorcery, Nodes of Yesod, Jack the Nipper, Camelot Warriors, etc. Busco copión.

José Luis García de la Fuente
Móstoles, 39, 3° Pta. 7
Tel. (91) 697 54 74
Fuenlabrada

Cambio juegos MSX. Tengo Soccer, Hyper Rally, Yie ar Kung Fu I y II, Boxing, etc. Escribir a:

Antonio Guillamó
Vicente Gallart, 24, 9°
46011 Valencia

Vendo consola videojuegos Philips con 14 cartuchos. Regalo juegos de TV Palson. Todo por sólo 15.000 ptas.

Ignacio Pérez Cuadrado
Tolosa Latour, 4, 6°C
Tel. 50 45 62
Cartagena (Murcia)

Deseo contactar para formar Club de MSX, para intercambio de ideas, juegos y programas.

Jon Masant
Apartado 7015
Bilbao

Cambio y vendo programas MSX, tengo entre otros: Hole in One, Tenis, Sky-Jaguar, Yie ar Kung-fu I y II. Escribir a:

Antonio Muñoz
Cuenca, bloque N, 2ª 1°
Montcada i Reixach (Barcelona)

Intercambio juegos MSX en cinta (Jet Fighter, Chess, 86, Vacuumania, Pyramid Warp, etc.). A ser posible con personas de Madrid.

Antonio
Tel. 269 06 47
Madrid

Compro ordenador MSX 80K. Si puede ser Philips por 30.000 ptas. Interesados llamar y dejar teléfono. Yo me pondré en contacto.

Juan Arboix Fajula
Paseo Valldaura, 150, piso 10, 6ª
Tel. 427 8917
08031 Barcelona

Cambio/Vendo programas MSX. Tengo: Road Fighter, Ping-Pong, Knight Lore, Hero, etc... Escribir a:

Antonio Bellido
Cuenca, bloque L, 1° 2ª
Montacada Reixach (Barcelona)

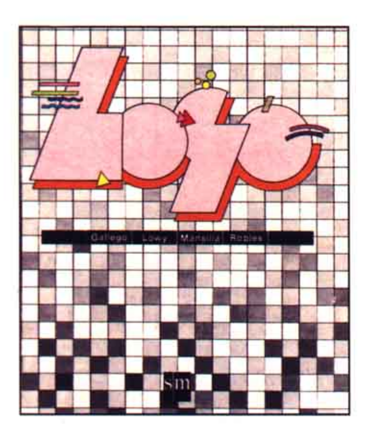
Vendo un buen desemsamblador, sólo ocupa 6,5 Kbytes. El cambio se hará contra reembolso. Precio a convenir. Quien quiera comprarlo que escriba a:

Oscar Saenz Maguregui
Escultor Francisco Durrio, 3, 8°C
48015 Bilbao

La redacción de Input
os desea unas



FELICES NAVIDADES



LOGO

Autor: Gallego, Lowy, Mansilla, Robles
Editor: Ediciones S.M.
Páginas: 269
Precio: N.D.

Logo es un texto encaminado a familiarizar a los potenciales alumnos con el lenguaje y el uso del ordenador a través de un lenguaje especialmente diseñado para estos fines: el LOGO.

Los objetivos del libro son esencialmente didácticos, con intención de ser usado en colegios, escuelas, etc. Por ello utiliza una versión del LOGO en castellano, y además se ciñe a las primitivas recomendadas por el M.E.C. y las comunidades autonómas.

Todo esto no quiere decir que no pueda utilizarse como un libro para uso personal. Todo lo contrario. La sencillez y claridad con la que están explicados los conceptos lo hacen muy adecuado para la consulta y aprendizaje personal.

Los temas se desarrollan suponiendo que van dirigidos a personas sin conocimientos ni experiencia en el mundo de los ordenadores. Los primeros capítulos están dedicados a los gráficos LOGO, explicando el manejo de la Tortuga (símbolo inequívoco de este lenguaje), y partiendo de ejemplos muy sencillos que se van complicando en capítulos posteriores.

A continuación se explican los procedimientos, con y sin paso de parámetros, para que el lector o alumno aprenda el diseño de sus propias aplicaciones mediante procedimientos simples, que es la idea base de la programación estructurada.

Se pasa a continuación a examinar el concepto de recursividad, o procedimientos que se llaman a sí mismos, explicando como puede hacerse uso de este concepto para resolver de forma más simple algunos problemas. Se aplican seguidamente los conceptos de procedimiento y de recursividad al manejo de listas, entendidas éstas como un conjunto de elementos que pueden ser caracteres, números, palabras u otras listas a su vez.

Por último, el libro termina con tres interesantes capítulos dedicados a la creación de gráficos sobre ejes cartesianos, a una profundización sobre el manejo de los gráficos-tortuga y a la creación de música.



MSX, APLICACIONES PARA LA CASA Y LOS PEQUEÑOS NEGOCIOS

Autor: J. Minguella
Editor: Noray S.A.
Páginas: 63
Precio: N.D.

Aparece en este libro un compendio de 15 programas desarrollados por el autor para nuestro ordenador **MSX**. Los temas sobre los que tratan los programas son de naturaleza diversa. Sin embargo, no hay ninguno de ellos que pueda calificarse realmente de novedoso.

Las aplicaciones están orientadas sobre todo a resolver con nuestro ordenador **MSX** los pequeños problemas domésticos y personales. Encontramos así un programa de gasto de combustible, un programa para enfrentarnos con el temido IVA, otro para llevar las cuentas bancarias, una contabilidad doméstica y dos agendas, una diaria para recordar lo que tenemos que hacer y otra telefónica.

Para tentar a la suerte tenemos un programa de quinielas y otro de lotería primitiva.

Dentro de la sencillez de las aplicaciones presentadas se ha intentado que la presentación de los programas en pantalla sea uniforme y agradable. También se ha uniformizado la entrada de datos.

Además se ha procurado, dentro de las limitaciones que el BASIC impone, que los programas tengan una estructura que sea lo más modular posible. Para ello los programas están divididos en un bloque de inicialización, un segundo bloque en el que se presenta el menú de opciones, y se hace la entrada de los datos y la comprobación de los mismos y un conjunto de bloques, que varían según las utilidades del programa (lectura y grabación de datos en cinta o en disco, presentación de resultados en pantalla o impresora, etc.)

Al principio se hace un breve comentario sobre el programa que se lista a continuación. Este comentario no explica la realización del programa, ni da instrucciones sobre su manejo, sino que no pasa de ser un breve comentario, a veces humanístico, sobre su utilidad.

MSX: INTRODUCCION PRACTICA AL SIMULADOR DE VUELO

Autor: M. Medel
Editor: Peopleware
Páginas: 179
Precio: N.D.

Este libro es la segunda parte de una obra del mismo autor, titulada **DESPEGA CON TU MSX**. En ella se nos enseñaban los comandos fundamentales del BASIC **MSX**, a la vez que se desarrollaba el diseño y puesta en vuelo de la nave espacial **Columbia**. Al final de aquella obra se dejaba para un libro posterior el diseño del cuadro de mandos interno y la creación de un pequeño simulador de vuelo de la nave.

Pues bien, cumpliendo con lo prometido, ya tenemos esa segunda parte en el mercado. La forma de desarrollar el contenido es muy parecida a la de la primera parte. Es decir, se divide el problema en bloques funcionales y se va tratando cada uno de ellos por separado. Sin embargo, hay una diferencia fundamental con respecto al libro anterior, y es que ya no se explican con detalle los comandos utilizados, más bien se hace hincapié en explicar la forma de hacer las cosas, las expresiones matemáticas que se utilizan y las bases físicas en que se fundamentan.

Incluso se dan también interesantes explicaciones sobre el diseño y funcionamiento de la nave **Columbia**, su forma de maniobrar y el cometido de cada uno de los instrumentos del cuadro de mandos. Todo ello se explica de forma amena y fácil de entender para cualquiera que posea unas nociones elementales del BASIC **MSX**. Siguiendo paso a paso los capítulos del libro dispondremos además de un estupendo simulador de vuelo con el que pasar ratos muy agradables y con la satisfación adicional de haberlo construido nosotros mismos paso a paso. Este simulador es muy completo, y tiene en cuenta efectos tales como la dependencia de la resistencia del aire con la altitud, las coordenadas geográficas (latitud y longitud), etc.



Philips New Media Systems

EL MAS OCUPADO DE LA CASA.

MSX DE PHILIPS.

Porque nadie puede resistirse a la tentación del MSX de Philips. A sus divertidos juegos de aventuras. A sus entretenidos programas educativos. O a los de oficina, como el "Home office". Capaz de hacer estadísticas, estudio de cuentas, contabilidad, etc.

Y los programas específicos para hacer más fácil el trabajo al ama de casa. O al estudiante. Además, posee una amplísima gama de periféricos: impresoras, monitores, ratón, etc. Disfrute con el MSX de Philips. Siempre que no esté ocupado.









Servicio de información al cliente: tels. (91) 413 21 61 - 413 22 46.